

DTIC FILE COPY

2

AD-A203 153



United States Army
Health Care Studies



and

Clinical Investigation Activity

SYSTEM DOCUMENTATION FOR THE U.S. ARMY
AMBULATORY CARE DATA BASE (ACDB) STUDY:
MAINFRAME, PERSONAL COMPUTER AND OPTICAL
SCANNER FILE STRUCTURE

GS-12 David R. Bolling
MAJ(P) Jeffrey P. Moon

Report HR89-0028

1 November 1988

DTIC
ELECTE
JAN 3 1 1989

S D
C D

US ARMY

DISTRIBUTION STATEMENT A

Approved for public release;
Distribution Unlimited

HEALTH SERVICES COMMAND
FORT SAM HOUSTON, TEXAS 78234

89 1 30 060

NOTICE

The findings in this report are
not to be construed as an official
Department of the Army position
unless so designated by other
authorized documents.

* * * * *

Regular users of services of the Defense Technical Information Center
(per DOD Instruction 5200.21) may purchase copies directly from the
following:

Defense Technical Information Center (DTIC)
ATTN: DTIC-ODR
Cameron Station
Alexandria, VA 22304-6145

Telephones: AUTOVON (108) 284-7633, 4, or 5
COMMERCIAL (202) 274-7633, 4, or 5

All other requests for these reports will be directed to the following:

U.S. Department of Commerce
National Technical Information Services (NTIS)
5285 Port Royal Road
Springfield, VA 22161

Telephone: COMMERCIAL (703) 487-4600

U

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION Unclassified			1b. RESTRICTIVE MARKINGS		
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION / AVAILABILITY OF REPORT Approved for public release; Distribution unlimited.		
2b. DECLASSIFICATION / DOWNGRADING SCHEDULE			5. MONITORING ORGANIZATION REPORT NUMBER(S)		
4. PERFORMING ORGANIZATION REPORT NUMBER(S) HR 89-002B			7a. NAME OF MONITORING ORGANIZATION DASG-RMP		
6a. NAME OF PERFORMING ORGANIZATION U.S. Army Health Care Studies & Clinical Investigation Acty		6b. OFFICE SYMBOL (if applicable) HSHN-H	7b. ADDRESS (City, State, and ZIP Code) 5109 Leesburg Pike Falls Church, VA 22041-3258		
6c. ADDRESS (City, State, and ZIP Code) Bldg 2268 Ft Sam Houston, TX 78234-6060		9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER			
8a. NAME OF FUNDING / SPONSORING ORGANIZATION HQDA(DASG-RMB)		8b. OFFICE SYMBOL (if applicable)	10. SOURCE OF FUNDING NUMBERS		
8c. ADDRESS (City, State, and ZIP Code) 5109 Leesburg Pike Falls Church, VA 22041-3258		PROGRAM ELEMENT NO.	PROJECT NO.	TASK NO.	WORK UNIT ACCESSION NO.
11. TITLE (Include Security Classification) (U) System Documentation for the U.S. Army Ambulatory Care Data Base (ACDB) Study: Mainframe, Personal Computer and Optical Scanner File Structure					
12. PERSONAL AUTHOR(S) GS12 David R. Bolling and MAJ(P) Jeffrey P. Moon					
13a. TYPE OF REPORT Final		13b. TIME COVERED FROM Jan 86 TO Sep 87		14. DATE OF REPORT (Year, Month, Day) 1988 November 1	
15. PAGE COUNT 695		16. SUPPLEMENTARY NOTATION (Defense Eligibility Enrollment System),			
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)		
FIELD	GROUP	SUB-GROUP	Ambulatory visits, DEERS, Diagnosis, Downloading data, FOCUS, Optical scanner, Personal computer, SAS, SIDPES, Uploading data.		
19. ABSTRACT (Continue on reverse if necessary and identify by block number) The U.S. Army Ambulatory Care Data Base (ACDB) Study was conducted at six medical treatment facilities during the period January 1986 - September 1987. This report outlines the optical scanning technology, personal computer and mainframe data processing support required for the project. Extensive coverage of the file structure and data element architecture developed to integrate a personal and mainframe data base management system is provided. Description of the FOCUS Data Base Management System programs, master files, and software profiles are included. Technical support was provided by the Data Processing Center, Fort Detrick, MD. Keywords: Medical computer applications,					
20. DISTRIBUTION / AVAILABILITY OF ABSTRACT <input type="checkbox"/> UNCLASSIFIED/UNLIMITED <input checked="" type="checkbox"/> SAME AS RPT. <input type="checkbox"/> DTIC USERS			21. ABSTRACT SECURITY CLASSIFICATION Unclassified		
22a. NAME OF RESPONSIBLE INDIVIDUAL David R. Bolling			22b. TELEPHONE (Include Area Code) (301) 663-7291		22c. OFFICE SYMBOL ASNE-HD-I

TABLE OF CONTENTS

ACKNOWLEDGMENTS	ix
INTRODUCTION	1.0
OPTICAL SCANNER	2.0
OPTICAL SCANNER OVERVIEW	2.1
DEVELOPMENT SCANNER	2.2
SCANNER CONSIDERATIONS	2.3
SCANNER PROGRAMMING	2.4
OVERVIEW	2.4.1
PROGRAMMING RESOLUTION SHEETS	2.4.2
OUTPUT DESIGN	2.4.3
DESCRIPTION OF THE SC PROGRAM	2.5
OPERATION OF THE SC PROGRAM	2.5.1
SC CODES	2.6
FORMAT OF SC FILES	2.7
LISTINGS OF SC FILES	2.8
COMPILING RESOLUTION SHEETS	2.9
COPYING SCANNER DISKETTES	2.10
DISTRIBUTION OF DISKETTES TO SITES	2.11
SCANNER SPEED	2.12
REFERENCES FOR SCANNER	2.13
SCANNER IBM XT SYSTEM	3.0
OVERVIEW	3.1
EQUIPMENT	3.2
FUNCTIONAL DESCRIPTION	3.3
PROGRAMMING DESIGN	3.3.1
INPUT RECORD	3.3.1.1
OUTPUT RECORD	3.3.1.2
PROGRAMS	3.4
OPERATION	3.4.1
OTHER PROGRAMS	3.5
COSORT	3.6
WARNING	3.6.1
SCANUTIL PROGRAM	3.7
MAIN MENU	3.7.1
CONFIGURATION MENU	3.7.2
RECORD FORMAT	3.7.3
REFERENCES	3.8

[illegible]

DATA BASE IBM XT SYSTEM	4.0
OVERVIEW OF DATA BASE IBM XT SYSTEM	4.1
EQUIPMENT	4.2
FUNCTIONAL DESCRIPTION	4.3
SOFTWARE	4.4
SYSTEM SOFTWARE	4.4.1
FOCUS PROGRAMS	4.5
OVERVIEW	4.5.1
PROFILE	4.5.2
MASTER FILES	4.6
VISIT MASTER FILE	4.6.1
PICTURE OF VISIT FILE	4.6.1.1
PATIENT MASTER FILE	4.6.2
PROVIDER MASTER FILE	4.6.3
CLINIC MASTER FILE	4.6.4
DIAGNOSIS MASTER FILE	4.6.5
PROCEDURE MASTER FILE	4.6.6
OTHER MASTER FILE	4.6.7
FIELD DESCRIPTIONS	4.7
VISIT FIELD DESCRIPTIONS	4.7.1
PATIENT FIELD DESCRIPTIONS	4.7.2
PROVIDER FIELD DESCRIPTIONS	4.7.3
CLINIC FIELD DESCRIPTIONS	4.7.4
DIAGNOSIS FIELD DESCRIPTIONS	4.7.5
PROCEDURE FIELD DESCRIPTIONS	4.7.6
OTHER FIELD DESCRIPTIONS	4.7.7
LOADING	4.8
REPORTING	4.9
CORRECTING	4.10
DUMPING	4.11
SPECIAL TOPICS	5.0
PATIENT REGISTRATION	5.1
SIDPERS	5.2
SIDPERS TAPE FORMAT	5.2.1
SIDPERS CONVERSION	5.2.2
SIDPERS OUTPUT RECORD	5.2.3
DEERS	5.3
DEERS TAPE FORMAT	5.3.1
DEERS FIELD DEFINITIONS	5.3.2
DEERS CONVERSION	5.3.3
DEERS PROCESSING	5.3.4

FT. DETRICK PROCESSING	6.0
PC PROCESSING AT DETRICK	6.1
UPLOADING	6.1.1
IRMA	6.1.1.1
DEVELOPMENT	6.1.2
TECHNICAL SUPPORT	6.1.3
OPTIMIS	6.1.3.1
MAINFRAME	6.2
TSO/ISPF ENVIRONMENT	6.2.1
FOCUS	6.2.2
FOCUS PROGRAMS	6.2.3
LOADING	6.2.3.1
REPORTING	6.2.3.2
RUNNING BATCH PROGRAMS	6.2.3.3
SAS PROGRAMS	6.2.4
DEERS AND SIDPERS PROCESSING	6.2.5
DOWNLOADING	6.2.6
SITE ACCESS TO MAINFRAME	6.3
MAINFRAME DATA BASE DESCRIPTION	6.4
MAINFRAME MASTER FILE DESCRIPTION	6.4.1
OLD VISIT MASTER FILE	6.4.1.1
CLINIC MASTER FILE	6.4.1.2
PATIENT MASTER FILE	6.4.1.3
PROVIDER MASTER FILE	6.4.1.4
PROCEDURE MASTER FILE	6.4.1.5
DIAGNOSIS MASTER FILE	6.4.1.6
NEW VISIT MASTER FILE	6.4.1.7
FIELD DESCRIPTIONS	6.4.2
PICTURE OF VISIT FILES	6.4.3
ENCRYPTION OF THE SSN	6.4.4

RESOLUTION SHEETS

APPENDIX A

SCAN400.FIL Adolescent Medicine
 SCAN410.FIL Allergy
 SCAN420.FIL Audiology/Speech
 SCAN430.FIL Cardiology
 SCAN440.FIL Cardiothoracic
 SCAN450.FIL Dermatology
 SCAN460.FIL Endocrine
 SCAN470.FIL ENT
 SCAN480.FIL Family Practice
 SCAN490.FIL Gastroenterology
 SCAN500.FIL General Surgery
 SCAN510.FIL Gynecology
 SCAN520.FIL Infectious Diseases
 SCAN530.FIL Internal Medicine
 SCAN540.FIL Nephrology/Dialysis
 SCAN550.FIL Neurology
 SCAN560.FIL Neurosurgery
 SCAN580.FIL Nutrition
 SCAN590.FIL OB
 SCAN600.FIL Oncology/Hematology
 SCAN610.FIL Ophthamology
 SCAN620.FIL Optometry
 SCAN630.FIL Ortho Appliance/Cast
 SCAN640.FIL Orthopedics
 SCAN650.FIL Occupational Therapy
 SCAN660.FIL Pain/Physical Medicine
 SCAN670.FIL Pediatrics
 SCAN680.FIL Physical Therapy
 SCAN690.FIL Plastic Surgery
 SCAN700.FIL Podiatry
 SCAN710.FIL Preventive Medicine
 SCAN720.FIL Primary Care
 SCAN730.FIL Psychiatry
 SCAN740.FIL Psychology
 SCAN750.FIL Pulmonary
 SCAN770.FIL Rheumatology
 SCAN780.FIL Social Work
 SCAN790.FIL Urology
 SCAN800.FIL BAS/TMC
 SCAN820.FIL EKG
 SCAN830.FIL Emergency Room
 SCAN840.FIL Group form I
 SCAN850.FIL Group form II
 SCAN860.FIL Immunization
 SCAN900.FIL Occupational Health
 SCAN910.FIL Occupational Therapy Repeat
 SCAN920.FIL Physical Therapy Repeat
 SCAN930.FIL Repeat
 SCAN940.FIL Short
 SCAN950.FIL Social Work Short

PC BASIC SCANNER PROGRAMS APPENDIX B

RACP200.LST Patient Form
 RACP300.LST Provider Form
 RACPGEN.LST General Form (400-790)
 RACP800.LST BAS/TMC Form
 RACP820.LST EKG Form
 RACP830.LST Emergency Room Form
 RACP850.LST Group I and II Forms
 RACP860.LST Immunization Short Form
 RACP900.LST Occupational Health Form
 RACP910.LST OT Repeat Form
 RACP920.LST PT Repeat Form
 RACP930.LST Repeat Form
 RACP940.LST Short Form
 RACP950.LST Social Work Short Form

PC BASIC SYSTEM AND MISC. PROGRAMS APPENDIX C

RACP05.LST Initial Sign-On
 RACP10.LST Initial Selection Menu
 RACP1A.LST OMR (Scanner) Select
 CONVERT SIDPERS Conversion

PC FOCUS PROCEDURES APPENDIX D

CLINIC.MAS Clinic Master file PC
 CLINIC1.FEX Report 1 Provider by Clinic
 CLINIC2.FEX Report 2 Patient Cat by Clinic
 CLINIC3.FEX Report 3 Referral Source
 CLINIC4.FEX Report 9 Job related
 DEERS.FEX DEERS Special Load PC
 DIAGDMP.FEX Diagnosis Description Dump PC
 DIAGNOS.MAS Diagnosis Master file PC
 LOADCLIN.FEX Clinic Description Load PC
 LOADDIAG.FEX Diagnosis Description Load PC
 LOADOTH.FEX Other Code Description Load PC
 LOADPROC.FEX Procedure Description Load PC
 MAIN.FEX Main menu program PC
 MILT1.FEX Report 10 Military Unit
 MILT2.FEX Report 11 Unit Health
 MILT3.FEX Report 12 (unfinished)
 OTHER.MAS Other Codes Master file PC
 PATIENT.MAS Patient Master file PC
 PROCDMP.FEX Procedure Description Dump PC
 PROCEDUR.MAS Procedure Master file PC
 PROFILE.FEX FOCUS Initialization file PC
 PROV1.FEX Report 4 Provider Diagnosis
 PROV2.FEX Report 5 Provider Procedure
 PROV3.FEX Report 6 Provider Time/Status
 PROV4.FEX Report 7 Provider Dx by Clinic
 PROVIDER.MAS Provider Master file PC
 RPTCRT.FEX Selection Menu for Rpts 1-6
 RPTCRT1.FEX Selection Menu for Rpts 7,8
 RPTCRT2.FEX Selection Menu for Rpts 10

RPTCRT3.FEX	Selection Menu for Rpts 11
RPTCRT4.FEX	Selection Menu for Rpts 12
RPTCRT5.FEX	Selection Menu for Rpts 9
RXXCCLD.FEX	Clinic Correction Dump PC
RXXCORL.FEX	Loads Corrected file to Visit
RXXCPAD.FEX	Patient Correction Dump PC
RXXCPRD.FEX	Provider Correction Dump PC
RXXF200.FEX	Patient Load file PC
RXXF300.FEX	Provider Load file PC
RXXF400.FEX	Visit Load file PC
RXXFCCL.FEX	Control Clinic Correction PC
RXXFCPA.FEX	Control Patient Correction PC
RXXFCPR.FEX	Control Provider Correction PC
RXXFDCL.FEX	Clinic Dump file PC
RXXFDPA.FEX	Patient Dump file PC
RXXFDPR.FEX	Provider Dump file PC
RXXFDVI.FEX	Visit Dump file PC
VISIT.MAS	Visit Master file PC

MAINFRAME FOCUS LOAD PROGRAMS APPENDIX E

CLINLOAD	Clinic Description
DIAGLOAD	Diagnosis Description
NLOAD	New Design Visit Data
OTHLOAD	Other Code Description
PASILO	Patient Sidpers
PATILOAD	Patient
PATILD	Patient Deers
PROCLOAD	Procedure Description
PROVLOAD	Provider
RXXF400	Old Design Visit Data

MAINFRAME SAS PROGRAMS APPENDIX F

DEERS	DEERS Conversion
PATD32	Patient Social Security Encryption
SIDCONV	SIDPERS Conversion

ACKNOWLEDGEMENTS

While a number of people contributed to the success of the Ambulatory Care Data Base study in the arena of system development, the site term employees contributions by far exceeded all others.

These persons, with little experience on personal computers, were the first to learn and operate the scanner system at each hospital. Their observations in processing the data and reporting problems were essential to the development effort. Long hours on the telephone were required in deciphering and correcting hardware and software errors.

The individuals are Sheron Jackson at Brooke Army Medical Center, Jennifer Gilmore at Ft. Bragg, Zoe Mitchell at Ft. Campbell, Nancy Smith and later Jeannette Ratcliffe at Ft. Jackson, Chuck Philpot at Redstone Arsenal and Walter Thompson at Ft. Polk. Their personal dedication to the project went considerably beyond their job description.

I would also like to thank Sue Akins for her efforts in editing this manual.

DRB

1.0 INTRODUCTION

This manual describes the Ambulatory Care Data Base (ACDB) Study data collection systems, system development techniques and mainframe procedures and programs used in maintaining and reporting ambulatory care data.

The many program listings and explanations are directed more for a programmer or systems analyst than a reader with little computer background. However, there are tables of the data structure and historical narrative that will aid in the data analysis. This manual does not contain operational descriptions of the systems since other manuals have addressed that topic.

Historical information is sometimes furnished to illustrate the reasons why some things were done and to point out problem areas. In some cases, warnings are given to avoid certain steps or processes should the method be used in a future study or standard system. Alternatively, the manual contains instructions for the use of various development tools and programs, descriptions of the programs used, and data structure from beginning to end.

The development process continued throughout the project. A major revision of all data collection forms required reprogramming all scanner and BASIC programs during the spring of 1987. In some cases, programs never reached completion as the data collection process ended rather abruptly in the summer of 1987. Thus, this manual is similar to a snapshot of where the project had been and was as of September 1987.

The development effort can be divided into four major areas. The first area pertained to the optical scanner which was used to read the raw data. The second area concerned the IBM XT system used to drive the scanner. This XT performed data conversions and edits while preparing data for the data base. The third major area involved the IBM XT which stored the data in a fourth generation data base management system (DBMS) called FOCUS. FOCUS was used to load data, write reports, prepare files for shipment, and perform other data maintenance activities. The fourth area related to the mainframe computer at Ft. Detrick. Here data from all the sites was collected and placed into cumulative files. The DBMS on the mainframe was also FOCUS, making many load and report programs similar to the PC versions. Operating procedures such as uploading data, processing Standard Installation Division Personnel System (SIDPERS) data, processing Defense Eligibility

Enrollment System (DEERS) data, loading data into FOCUS and downloading patient data to be sent back to the sites are described within this document.

An entire document could be written about end user support but little is discussed here. Ft. Detrick provided technical support to all sites throughout the project. The hospital site temporary employees had little or no computer experience. They quickly learned to follow detailed directions and also provided feedback when problems occurred. Their support allowed for rapid resolution of problems, thus preventing major disruptions. As a result, these temporary employees were instrumental to the success of the study.

Lastly, the project used various languages and equipment with specialized documentation. This manual does not duplicate any documentation already written but provides references to the documentation. In some cases, additional explanations are furnished where external documentation was lacking or insufficient.

2.1 OPTICAL SCANNER OVERVIEW

The ACDB study team designed optical scanning forms for each outpatient specialty or clinic. The forms were filled out (bubbled in) by clerks and/or providers in each clinic. Completed forms were taken to a central location where they were read by a National Computer Systems (NCS) scanner (Models 7001-7006). Some editing of the forms was performed through a dedicated computer within the scanner. The scanners at each site were connected to an IBM XT personal computer which controlled the scanner as well as performed additional editing functions. Forms were accepted or rejected by the scanner. Accepted forms were directed to a hopper on the scanner, and the data was decoded and saved in a file on the hard disk drive of the IBM computer. Rejected forms were directed to a reject hopper where they could be corrected and rescanned. Correction sometimes involved sending the forms back to the originating clinic for further information.

2.2 DEVELOPMENT SCANNER

The scanner used for development was an NCS model 7001.

2.3 SCANNER CONSIDERATIONS

The scanner read both sides of the forms simultaneously by using an optical read head that shone through the paper. This method required the forms to be designed so that no two marks were positioned back-to-back. Stray marks carelessly written on one side of the form could be read as an input value on the opposite side. Also, special paper had to be used.

2.4 SCANNER PROGRAMMING

2.4.1 OVERVIEW

The scanner computer was programmed by bubbling in resolution sheets. Each resolution sheet (see sample of blank sheet) had nine boxes where coordinates and other information about the type of input could be entered. Ten to twenty resolution sheets were required for each form. The resolution sheets were compiled by using the scanner compilation utility. The compiled program was stored on a diskette which resided in the scanner during scanning of forms.

2.4.2 PROGRAMMING RESOLUTION SHEETS

Details for programming the resolution sheets are provided in the User's Guide for the Sentry 70 Scanners and thus will not be covered here. The process of coding the resolution sheets was tedious and subject to errors. One bubble error in a 20 sheet program could cause considerable debugging time.

There were numerous data fields on the ACDB forms that were identical from form to form. The forms were designed so that the identical fields were located in the same coordinates on every form where possible. In this way many resolution sheet boxes could be duplicated from program to program. Manual copying of the resolution sheets by clerks introduced an average of one or more errors per page, and the debugging time negated any time saved by this method.

NCS supplied a program for the IBM PC that allowed entering the resolution sheet data at a computer terminal. The program printed out (preslugged) continuous resolution sheets using special printer ribbons. This program called SC.EXE made the coding and debugging of scanner programs more efficient.

2.4.3 OUTPUT DESIGN

The scanner in operation sent an output record to the IBM XT. This record served as input to the Basic programs, and, therefore, details of its format were essential. Records were designed to be as consistent as possible to make programming and testing easier. The following table lists the fields of the scanner record and their positions:

SCANNER OUPUT RECORD FORMAT

DESCRIPTION	FORM TYPE			SIZE	POSITION	NOTES
	1	2	3			
HEADER	X	X	X	21	1-21	Provided by scanner
LITHO	X	X	X	8	22-29	
CLINIC PFX	X	X	X	1	30	(1-5)
CLINIC CODE	X	X	X	3	31-33	Alphabetic
OTHER UCA	X	X	X	1	34	No. of responses may vary
TODAYS DATE	X	X	X	4	35-38	MoDa
PROV 1 PFX	X	X	X	1	39	Alphabetic
PROV 1 CODE	X	X	X	4	40-43	Numeric
SSN	X	X	X	9	44-52	Numeric
FMP	X	X	X	2	53-54	Numeric
VISIT COUNT	X	X	X	1	55	(1-9) 1 Counts as zero
PROV 1 TIME	X	X	X	2	56-57	(1-22)
PROV 2 PFX	X	X	X	1	58	Alphabetic
PROV 2 CODE	X	X	X	4	59-62	Numeric
PROV 2 TIME	X	X	X	2	63-64	(1-22)
REASON FOR #2	X	X	X	1	65	(1-6) (Extra bubble on psy)
IF NOT SCHED	X	X	X	1	66	(1-2)
IF NOT CLINIC	X	X	X	1	67	(1-6)
REF CODE PFX	X	X	X	1	68	(1-6)
REF CODE	X	X	X	3	69-71	Alphabetic
JOB RELATED	X	X	F	1	72	F - filler
MIL ONLY DUTY	X	X	F	1	73	(1-2)
MIL ONLY QTRS	X	X	F	1	74	(1-3)
MIL ONLY PROF	X	X	F	1	75	(1-4)
FILLER	F	F	F	5	76-80	Used for ref codes on 65,68
SPEC PRE CLIN	X	X	X	9	81-89	Array
NOT AVAILABLE	X	X	X	3	90-92	Array
PROV 2 ADDL I	X	X	X	1	93	(1) zero by default
ADDL PROC I	X	X	X	5	94-98	Numeric
PROV 2 ADDL II	X	X	X	1	99	(1) zero by default
ADDL PROC II	X	X	X	5	100-104	Numeric
ADMITTED	X	X	X	1	105	(1) May be filler
UNL REAS PRIM	X	X	X	1	106	(1-2) Prefix
UNL REAS PRIM	X	X	X	5	107-111	Dx code
UNL REAS SEC	X	X	X	1	112	(1-2) Prefix
UNL REAS SEC	X	X	X	5	113-117	Dx code
FOLLOW/R OUT	X	X	X	1	118	(1-2)
PROC PROV 1	X	X	X	N	119-118+N	Array with N procedures
PROC PROV 2	X	X	X	N	119+N-118+2N	Array with N procedures
OTHER CODES	X	X	X	P	119+2N-118+2N+P	Array with N codes
DX 1 COL 1	X	X	X	2	119+2N+P-120+2N+P	Col position
DX 1 COL 2	X	X	X	2	121+2N+P-122+2N+P	Col position
DX 1 COL 3	X	X	X	2	123+2N+P-124+2N+P	Col position
DX 1 COL 4	X	X	X	2	125+2N+P-126+2N+P	Col position
DX 2	X	X	X	M	127+2N+P-126+2N+M+P	Array with M dx
PSYCHOMETRIC Only on Psycol				4	127+2n+P+M-130+2n+M+P	Array
SPEC PROGRAMS	F	X	F	7		Array (note fillers)

The forms were grouped into three categories. All forms in each category had the same number of fields. When a form did not have a field, a blank or filler was provided on the scanner output record. These are indicated by an F in the table. Fields of varying length, such as the one for procedures, were placed near the end of the record.

2.5 DESCRIPTION OF THE SC PROGRAM

The SC.EXE program provided by NCS to preslug resolution sheets was not a standard program at that time. As a result, little documentation was provided. Operation of the program had to be determined; therefore, some details in the operation of the SC program are provided here.

2.5.1 OPERATION OF THE SC PROGRAM

The SC program could be loaded onto a floppy diskette or hard disk and called by typing 'SC'. The program first requested an output file name. This was the name of a file that contained the data needed to preslug the resolution sheets. This file could be copied using DOS. The SC program allowed an option to edit fields or add new resolution sheet boxes. Correcting or modifying the file could also be done by an independent editor. In fact, an independent editor was necessary since the DISPLAY field was not handled by the SC program. The DISPLAY field provided the scanner with the box containing the form Litho or serial number information. This Litho number could then be sent to the IBM computer to be used as an identifier in error listings.

Although external editing of the SC files was permitted, care had to be taken as the SC program did some things automatically. One such process was the calculation of the character count for each form. The character count was calculated and placed in the last line of the file. Also, the character count had to be edited if any box was modified in such a way as to change the character count.

WARNING: The SC program is not sophisticated in reading its own files. It determines lines by counting characters. Thus every line in an SC file must be 46 characters long. Some editors may remove trailing blanks on a line, causing the SC program to misread the file.

The SC program is menu driven and, for one with some knowledge of resolution sheet programming, is fairly user friendly. Some fields are coded, and these are listed below to permit reading of the SC output files contained in this documentation.

2.6 SC CODES

The following codes were used in the SC program when entering resolution sheet information. These codes were provided on the screen menu during operation.

<u>CLASS</u>	<u>CODE</u>	<u>EDIT</u>	<u>CODE</u>	<u>S</u>	<u>10</u>
A	0	C	1	N	11
N	1	C/B	2		
R	2	M	3		
RD	3	LJ	4		
B	4	RJ	5		
BD	5	10	6		
D	6	PY	7		
L	7	PS	8		
F	8	L	9		
				<u>IF ERROR</u>	<u>CODE</u>
				STOP	1
				FLAG	2
				SEL&REJ	3
				SEL&FLAG	4

2.7 FORMAT OF SC FILES

There were several types of records: Header, Box, Filler, and Trailer. Formats of each type are given below:

HEADER RECORD 1

<u>POSITION</u>	<u>DESCRIPTION</u>
1	Record type H
2-3	Skunk Mark count
4-5	Timing Mark count
6-9	Bias Bar (Track and Cell count)
10	Booklet Sequence Check
11	Output Control Fields
12-14	Display Page
15	Display Box
16-46	Program Notes

HEADER RECORD 2

<u>POSITION</u>	<u>DESCRIPTION</u>
1	Record type H
2-3	First Skunk Mark Position
4-5	Second Skunk Mark Position
Additional Skunk Mark Positions as needed.	

BOX RECORD

<u>POSITION</u>	<u>DESCRIPTION</u>
1	Record type B
2-19	Box Notes
20	Class
21-22	X Starting Coordinate
23-24	X Ending Coordinate
25-26	X Spacing
27-28	Y Starting Coordinate
29-30	Y Ending Coordinate
31-32	Y Spacing
33-34	Edits
35	If Error
36	Horizontal/Vertical
37-40	Number of Responses in Box
41-42	Number of Characters
43-46	Box Number

FILLER RECORD (Leaves an unused box)

<u>POSITION</u>	<u>DESCRIPTION</u>
1	Record type F
Remaining positions same as BOX RECORD except that positions 2-42 are ignored by the program.	

TRAILER RECORD

<u>POSITION</u>	<u>DESCRIPTION</u>
1-5	Contains //EOF
6-9	Character count

2.8 LISTINGS OF SC FILES

Documentation of resolution sheets is provided by listing the SC files since they take up less space than the resolution sheets. The SC files are in Appendix A.

2.9 COMPILING RESOLUTION SHEETS

Compilation of the resolution sheets was done on the NCS 7000 series scanners. An 'IDENTIFICATION SHEET' form was placed at the beginning of the resolution sheets for each program. The beginning form had to be coded with the program number and a character count. The character count could be obtained from the last line of the SC file or by manually summing the character responses for each box on the resolution sheets. Care should be taken not to confuse response count with

character count. The following example explains the difference. In reading a Social Security number of nine digits, nine columns of bubbles are required. Each column must have ten digits, 0 through 9. Thus there are 90 responses but the final result is nine characters. The nine characters are what is counted for the character count.

The character count is important in that the program will not compile unless this number is correct. An 'OPERATION HEADER/TRAILER' form follows each set of resolution sheets.

The scanner was turned on and set to the COMPILE mode using the select button. A properly formatted diskette was inserted into the disk drive of the scanner. The resolution sheets with the IDENTIFICATION sheet and TRAILER sheet were placed face down in the input hopper. The start button was pressed. The compiled program was then written to the diskette. The compiler detected some errors as each sheet was scanned. The sheets could be corrected and placed back into the input hopper without restarting. If the character count was incorrect, determination could be made to see if the count was incorrect or if a box was coded improperly, resulting in a false count. If the program compiled correctly, the last sheet hesitated several times before being released to the output hopper.

Compilations were slow, and debugging could be tedious. The preslugged forms could be misread if a slight misalignment occurred or if the ribbon was worn or shelf life had exceeded six months. The scanner had to be properly maintained to read the resolution sheets efficiently. Some hints to make this operation easier are (1) make sure the printer ribbon is new and the forms are aligned properly, (2) wipe the read heads on the scanner to insure they are clean, (3) vacuum any dust that may be present in the scanner read path, (4) make sure the environment of the scanner and forms has low humidity. Should a form be misread because of light bubbles, darken the bubbles with a pencil and rescan.

2.10 COPYING SCANNER DISKETTES

Once programs were compiled, they resided on several diskettes. The number of programs on one diskette depended on the length of the program. About 12-14 programs fit on one diskette based on the forms used in the ACDB Study. These diskettes had to be copied and sent to each site involved in the study. A

diskette copy utility was available on the scanner. However, because there was only one drive and because the scanner's computer memory was small, several exchanges of the source diskette and target diskette were necessary.

2.11 DISTRIBUTION OF DISKETTES TO SITES

Copies of the scanner program diskettes were sent by mail to each site involved with the study. Corrections to any program on a diskette required a new copy of the entire diskette to be sent out. An exception to copying diskettes for all sites is discussed in section 2.12.

2.12 SCANNER SPEED

One of the several features of the 7006 model scanner was that it could scan forms at a faster rate than the 7001 or 7004 models. To do this, however, the programs had to be compiled on a 7006 scanner. To achieve the maximum scan rate, the resolution forms were sent to the Brooke Army Medical Center (BAMC) system administrator. The programs were compiled on the 7006 scanner at BAMC and copies were sent to all sites having the 7006 scanner.

2.14 REFERENCES

BASIC/PASCAL Tabletop Utilities User's Guide for the Tabletop Scanners. (1984). Minneapolis, MN: National Computer Systems, Inc.

Operations Self Instruction Manual for the Tabletop Scanners. (1983). Minneapolis, MN: National Computer Systems, Inc.

User Reference Manual for the Tabletop Scanners. (1984). Minneapolis, MN: National Computer Systems, Inc.

3.0 SCANNER IBM XT SYSTEM

3.1 OVERVIEW OF SCANNER IBM XT SYSTEM

An IBM XT system was used in conjunction with an NCS scanner to read the ACDB forms and create a data set containing the output data.

The first system had only one XT which also served as a data base management system using FOCUS. The data base management system required so much time in loading, dumping, reporting and other operations such as report program development that there was not enough time left to scan forms. Thus after a year of operation, a second XT, which had a specific task of controlling the scanner and preparing the data for loading into the DBMS was added to the system. The second XT was added to all sites except Redstone where the quantity of data was smaller than at the other sites. This exception added to the development work in that a new system was added to the extra XTs while Redstone had to remain on the old system. Later modifications had to be specially customized for Redstone.

3.2 EQUIPMENT

The additional IBM XT was installed at all sites except Redstone during the months of October and November 1986. The system consisted of an IBM XT with a monochrome monitor and 256K of memory, an AST Sixpakplus board with 384K of memory, which brought the total memory to 640K, and an Iomega Bernoulli 20M byte single disk drive. DOS Version 3.1 was installed.

3.3 FUNCTIONAL DESCRIPTION

The NCS scanners were connected to an IBM XT system using an RS232 cable via the second serial port. Although the scanner had a built-in computer that performed some editing, the XT could perform more sophisticated editing. This editing was done in several programs written in IBM Compiled Basic. These programs read each scanner output record, edited the data, notified the scanner when the data was unacceptable, converted data, performed table lookups and created an ASCII file as output. Data in the output file could be accumulated until the operator loaded it into the data base. The output file was then dumped onto a 20M byte Iomega cartridge which could be moved to the data base management system for loading.

A login procedure was necessary to prevent

unauthorized access to the system. After logging on, the operator was guided through the options available on the system by a system of menus.

3.3.1 PROGRAMMING DESIGN

The Basic programs to edit the scanner output record were written to be as generic as possible. The first phase of the study had a Basic program for every form. After the redesign of the forms, one program was written to edit forms 400 through 790. Writing and testing one program was less time consuming, and maintenance was easier. Previously, when a universal change had to be made, all programs had to be modified, tested, and sent out to the study sites. Combining a number of forms into one general program did have risks, however, since a change to fix a problem on one form could affect others without it being known. Thorough testing was required to prevent this kind of error.

3.3.1.1 INPUT RECORD

After reading a form, the scanner sent the output record to the IBM XT. Details of the scanner output record are in Section 2.4.3. In the XT a Basic program read, decoded and edited the scanner record. General algorithms were written in the Basic programs to read and decode various sections of a form such as the procedure list and diagnosis list. The following table contains parameters used in these algorithms:

PARAMETERS FOR BASIC PROGRAMS

NO.	FORM	TYPE	REC LEN	NO. PROC	NO. OTHR	NO. COL	1DX COL	2DX COL	3DX COL	4DX COL	DX2 OFF
200	PATIENT REGISTRATION	U									
300	PROVIDER REGISTRATION	U									
400	ADOLESCENT MEDICINE	3	303	28		4	34	35	30	24	-2
410	ALLERGY	1	276	43		3	17	19	31		-1
420	AUDIOLOGY/SPEECH	1	291	49		3	31	28	11		-1
430	CARDIOLOGY	1	323	29		4	43	34	33	30	-1
440	CARDIOTHORACIC SURGERY	1	276	15		4	37	28	28	27	0
450	DERMATOLOGY	1	466	92		4	42	43	44	28	-1
460	ENDOCRINE	1	205	10		3	18	18	26		-1
470	ENT	1	305	41		4	25	37	20	16	-1
480	FAMILY PRACTICE	1	315	16		4	48	41	42	28	-2
490	GASTROENTEROLOGY	1	293	29		3	33	40	39		-1
500	GENERAL SURGERY	1	322	36		4	26	35	42	22	-1
510	GYNECOLOGY	1	406	58		4	44	44	46	32	-2
520	INFECTIOUS DISEASE	1	230	17		3	30	24	19		-1

NO.	FORM	TYPE	REC LEN	NO. PROC	NO. OTHR	NO. COL	1DX COL	2DX COL	3DX COL	4DX COL	DX2 OFF
530	INTERNAL MEDICINE	1	230	13		4	25	27	18	10	-2
540	NEPHROLOGY	1	380	64	24	4	31	37	18	17	-1
550	NEUROLOGY	1	371	17		4	59	62	50	41	-1
560	NEUROSURGERY	1	393	20	29	4	60	62	61	17	-2
570	NUCLEAR MEDICINE	1									
580	NUTRITION CARE	3	310	38		3	42	38	32		-2
590	OBSTETRICS	1	307	44		4	25	24	34	12	-2
600	ONCOLOGY/HEMATOLOGY	3	321	24		4	40	35	39	34	-1
610	OPHTHAMOLOGY	1	314	41		4	30	28	32	18	-2
620	OPTOMETRY	1	353	51	21	4	34	30	26	16	-2
630	ORTHO APPLIANCE/CAST	3	334	41	107	2	12	11			0
640	ORTHOPEDICS	1	309	25		4	35	43	33	23	-1
650	OCCUPATIONAL THERAPY	1	469	120		4	28	31	31	15	-2
660	PAIN/PHYSICAL MEDICINE	1	325	50		3	38	43	21		-1
670	PEDIATRICS	3	336	40		4	25	45	37	25	-2
680	PHYSICAL THERAPY	1	373	65	19	4	37	34	17	12	-2
690	PLASTIC SURGERY	1	270	22		4	22	27	26	26	-1
700	PODIATRY	1	324	51		3	18	42	39		-1
710	PREVENTIVE MEDICINE/CHN	1	351	55		4	32	31	34	20	-2
720	PRIMARY CARE	1	254	12		4	34	29	28	15	-2
730	PSYCHIATRY	2	362	10	30	4	60	52	55	14	-2
740	PSYCHOLOGY (4 PSY ASSM)	2	396	38	34	3	52	40	59		0
750	PULMONARY	1	285	36		3	40	25	25		-1
760	RADIOTHERAPY	3									
770	RHEUMATOLOGY	1	284	9	11	4	36	32	34	28	-1
780	SOCIAL WORK	2	317	34	1	4	30	35	29	22	-1
790	UROLOGY	1	324	56		4	19	23	33	12	-1
800	BAS/TMC	U	330	25		4	37	30	42	40	-2
810	EFMP	U									
820	EKG	U	239	18							
830	EMERGENCY ROOM	U	461	71		4	55	63	50	26	-1
840	GROUP FORM I	U	95	98							
850	GROUP FORM II	U	194								
860	IMMUNIZATION SHORT	U	397	39							
870	INDIVIDUAL MEDICAL READ	U									
880	INHALATION THERAPY	U									
890	NUTRITION CARE SHORT	U									
900	OCCUPATIONAL HEALTH	U	534	81	20	4	33	36	39	26	-1
910	OCCUPATIONAL THER REP	U	352	26							
920	PHYSICAL THERAPY REPEAT	U	352	26							
930	REPEAT	U	118								
940	SHORT	U	252	16							
950	SOCIAL WORK SHORT	U	183	5							

The above table was constructed during program development and can be used as a guide for developing new forms or debugging problems. An explanation of the table follows:

Program Number

The first column contains an arbitrary program number. This number is used by the scanner and XT to identify programs and tables belonging to each form.

Form

The second column provides the form description.

Form Type

The form type generally categorizes the forms into similar groups based on the design of the form. There are three general types (1-3) and a number of unique types designated by 'U'.

Record Length

The record length shown in the fourth column is the most important parameter in the table. It is the length of the scanner record being sent to the XT. The length includes 21 characters in a header plus the character count from the scanner program. A formula used to determine the length of the record based on the other parameters for the general form types is given below:

$$L = 118 + 2N + P + 2C + M + F$$

where L is the record length
N is the number of procedures
P is the number of other codes
C is the number of diagnosis columns
M is the total number of diagnoses
F is the Dx2 offset

Each scanner program was designed to output an identical section of the record which totaled 118 characters. Blank fillers were used for those fields not present on a form.

Number of Procedures

The fifth column of the table lists the number of procedures present on each form.

Number of Other Codes

Several forms have sections that are similar to the procedure list but are not procedures. They have been given the generic name of 'Other Codes.' Column six in the table shows the number of 'Other Codes' on each form.

Number of Columns

The diagnosis section on each form is broken up into columns. The seventh column in the table indicates the number of columns on each form.

Number of Diagnoses in a Column

The four columns labeled 1DX COL, 2DX COL, 3DX COL and 4DX COL contain the number of diagnoses in each column present on a form.

Secondary Diagnosis Offset

There are generally two bubbles beside each diagnosis description on a form, one for the primary diagnosis, and one for the secondary diagnosis. Where the last one or two diagnoses indicate 'no problem noted' or some similar description, no secondary bubble is present. The last column of the table indicates the number of entries for diagnoses that have no secondary bubble. In other words, the total number of primary diagnoses on a form would be the sum of the entries from the 1DX, 2DX, 3DX, and 4DX columns, and the total number of secondary diagnoses would be the same number plus the offset.

3.3.1.2 OUTPUT RECORD

After a form was read by the scanner and edited by a Basic program it was either accepted or rejected. The rejected forms were sent to a reject hopper on the scanner and could be corrected by the study administrator or sent back to the clinic. The Basic program provided a listing of the serial numbers (litho code) of the rejected forms and described the errors found on each form.

The data from accepted forms was then written to an ASCII file for later inclusion in the FOCUS data base. Several types of records were written so FOCUS could load the data into its hierarchical structure. The following table describes the formats of these output records:

BASIC PROGRAM OUTPUT RECORD FORMAT

RECORD TYPE 1 GENERAL WITH PRIMARY DX

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS 1
CLINIC ID	4	5-8	
VISIT DATE	6	9-14	
PRIM PROV	5	15-19	
SSN	9	20-28	
FAM MEM PFX	2	29-30	
LITHO	8	31-38	
FORM NUMBER	2	39-40	
VISIT COUNT	1	41	
PRI PROV TIME	3	42-44	
SEC PROV	5	45-49	
SEC PROV TIME	3	50-52	
REAS FOR SEC PROV	1	53	
APPOINTMENT STAT	1	54	
REFERRAL CODE	4	55-58	
PLACE OF VISIT	1	59	
JOB REL VISIT	1	60	
MIL DUTY	1	61	
MIL QTR	1	62	
MIL PROFILE	1	63	
NOT-AVAILABLE	1	64	
ADMITTED	1	65	
INFIELD	1	66	
INJURIES	1	67	
PURPOSE OF VISIT	1	68	
PRIM DX RULE OUT	1	69	
PRIM DX	5	70-74	
PADDING	6	75-80	

RECORD KEY
COMMON TO ALL
RECORD TYPES

RECORD TYPE 2 PROCEDURE CODE

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS 2
RECORD KEY	36	5-40	COMMON TO ALL RECORD TYPES
PROV INDICATOR	1	41	
PROCEDURE CODE	5	42-46	
PADDING		47-80	

BASIC PROGRAM OUTPUT RECORD FORMAT

RECORD TYPE 3 SPECIFIC PREASSIGNED CLINIC CODES

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS 3
RECORD KEY	36	5-40	COMMON TO ALL RECORD TYPES
SPE PRE CLINIC CD 1		41	
PADDING		42-80	

RECORD TYPE 4 OTHER CODES

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS 4
RECORD KEY	36	5-40	COMMON TO ALL RECORD TYPES
OTHER CODES	5	41-45	
PADDING		46-80	

RECORD TYPE 5 GROUP SEGMENT

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS 5
RECORD KEY	36	5-40	COMMON TO ALL RECORD TYPES
UNIT ID CODE	6	41-46	
TIME PREP PROV 1	3	47-49	
TIME PREP PROV 2	3	50-52	
TIME TRAV PROV 1	3	53-55	
TIME TRAV PROV 2	3	56-58	
NO. ACT DUTY	3	59-61	
NO. OTHER DUTY	3	62-64	
NO. RETIRED MIL	3	65-67	
NO. DEPENDANTS	3	68-70	
NO. CIVILIANS	3	71-73	
CONTINUATION	1	74	
PADDING		75-80	

BASIC PROGRAM OUTPUT RECORD FORMAT

RECORD TYPE 6 SPECIAL PROGRAMS

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS 6
RECORD KEY	36	5-40	COMMON TO ALL RECORD TYPES
SPECIAL PROGRAMS	1	41	
PADDING		42-80	

RECORD TYPE 7 MULTIPLE SECONDARY DX

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS 7
RECORD KEY	36	5-40	COMMON TO ALL RECORD TYPES
SEC DX	5	41-45	
PADDING		46-80	

RECORD TYPE 8 PSYCHOMETRIC ASSESSMENT

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS 8
RECORD KEY	36	5-40	COMMON TO ALL RECORD TYPES
PSYCHOM ASSESS	1	41	
PADDING		42-80	

RECORD TYPE 9 DISPOSITIONS (FROM OCC HEALTH FORM)

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS 9
RECORD KEY	36	5-40	COMMON TO ALL RECORD TYPES
DISPOSITION	2	41-42	
PADDING		43-80	

BASIC PROGRAM OUTPUT RECORD FORMAT

RECORD TYPE A GROUP II DATA

<u>FIELD</u>	<u>SIZE</u>	<u>POSITION</u>	<u>NOTES</u>
PROGRAM ID	3	1-3	
RECORD TYPE	1	4	CODED AS A
RECORD KEY	36	5-40	COMMON TO ALL RECORD TYPES
SSN	9	41-49	
FMP	2	50-51	
PADDING		52-80	

3.4 PROGRAMS

The programs in the IBM XT Scanner System were written in the BASIC language for the IBM Basic Compiler Version 1.0.

A list of compiled BASIC programs and their functions follows:

RACP05.EXE	Initial sign on program.
RACP10.EXE	Main menu program.
RACP1A.EXE	Scanner form program select.
RACPGEN.EXE	General decode program for forms 400-790.
RACP200.EXE	Decode prog. for form 200 (Patient reg.)
RACP300.EXE	Decode prog. for form 300 (Provider reg)
RACP800.EXE	Decode prog. for form 800 (BAS/TMC)
RACP820.EXE	Decode prog. for form 820 (EKG)
RACP830.EXE	Decode prog. for form 830 (Emergency rm)
RACP850.EXE	Decode prog. for forms 840-850 (Group)
RACP860.EXE	Decode prog. for form 860 (Immun. Short)
RACP900.EXE	Decode prog. for form 900 (Occ Health)
RACP910.EXE	Decode prog. for form 910 (OT Repeat)
RACP920.EXE	Decode prog. for form 920 (PT Repeat)
RACP930.EXE	Decode prog. for form 930 (Repeat)
RACP940.EXE	Decode prog. for form 940 (Short)
RACP950.EXE	Decode prog. for form 950 (Soc Wk Short)

Compiled programs all have the EXE extension. The source programs have a BAS extension. Listings of the compiled programs have an LST extension. Program listings can be found in the appendix. Compiled programs may contain subroutines that are common to more than one program. While specific subroutine listings are not provided in this documentation, the compiled versions of the subroutines are included with each program listing. The names of the subroutines and their

functions are shown below.

RACDIM.MOD	Dimension Definitions.
RACCMN.MOD	Common Area Definitions.
RACDEF.MOD	Default Definitions.
RACS1000.SUB	Logon verification.
RACS2000.SUB	Wait and Reply.
RACS5000.SUB	Date editor
RACS5700.SUB	Map ones to position.
RACS5800.SUB	Four col Dx converter.
RACS6000.SUB	Instring search.
RACS7000.SUB	Screen header display.
RACS7100.SUB	Print standard heading.
RACS8000.SUB	Scanner decoder.
RACS9000.SUB	Scanner I/O controller.
RACS9010.SUB	Scanner control options.
RACS9020.SUB	Scanner sheet call.
RACS9030.SUB	Scanner transport print call.

Two other subroutines used in programs reading forms were RACMnnn.MO1 and RACMnnn.MO2 where nnn is the form number. The corresponding subroutine names for the general form program are RACREAD.MOD and RACWRIT.MOD. The first subroutine decoded and converted data as well as performed editing functions. The second subroutine wrote the output records to an ASCII file.

The procedure and diagnosis codes for each form were typed into separate files and were read by the program to be used for table lookups. Each file began with a blank code and ended with a code of five 'Z's. All codes were five characters and enclosed in quotes. Diagnosis codes less than five characters were padded with blanks. Each file had an extension corresponding with the form number. Files were named RACDIAG or RACPROC depending on whether they were diagnosis codes or procedure codes. The names of all table files are

RACDIAG.400	RACDIAG.620	RACPROC.400	RACPROC.620
RACDIAG.410	RACDIAG.630	RACPROC.410	RACPROC.630
RACDIAG.420	RACDIAG.640	RACPROC.420	RACPROC.640
RACDIAG.430	RACDIAG.650	RACPROC.430	RACPROC.650
RACDIAG.440	RACDIAG.660	RACPROC.440	RACPROC.660
RACDIAG.450	RACDIAG.670	RACPROC.450	RACPROC.670
RACDIAG.460	RACDIAG.680	RACPROC.460	RACPROC.680
RACDIAG.470	RACDIAG.690	RACPROC.470	RACPROC.690
RACDIAG.480	RACDIAG.700	RACPROC.480	RACPROC.700
RACDIAG.490	RACDIAG.710	RACPROC.490	RACPROC.710
RACDIAG.500	RACDIAG.720	RACPROC.500	RACPROC.720
RACDIAG.510	RACDIAG.730	RACPROC.510	RACPROC.730
RACDIAG.520	RACDIAG.740	RACPROC.520	RACPROC.740
RACDIAG.530	RACDIAG.750	RACPROC.530	RACPROC.750

RACDIAG.540	RACDIAG.770	RACPROC.540	RACPROC.770
RACDIAG.550	RACDIAG.780	RACPROC.550	RACPROC.780
RACDIAG.560	RACDIAG.790	RACPROC.560	RACPROC.790
RACDIAG.580	RACDIAG.800	RACPROC.580	RACPROC.800
RACDIAG.590	RACDIAG.830	RACPROC.590	RACPROC.830
RACDIAG.600	RACDIAG.900	RACPROC.600	RACPROC.900
RACDIAG.610		RACPROC.610	

3.4.1 OPERATION

The operation of the Scanner XT System began by invoking a DOS batch file called STRTACDB. The FOCUS XT System began with a batch file of the same name. Each system could be configured by the AUTOEXEC.BAT file to move to the proper directory and automatically call the STRTACDB batch file. For those familiar with DOS, the operator could change directories and type STRTACDB to enter the corresponding system. In the Scanner system the STRTACDB batch file was as follows:

```
ECHO OFF
C:
CD \AMB
PATH C:\AMB;C:\;D:\
CLS
RXXP05
RXXB00
C:
CD \
CLS
```

The scanner programs resided in the AMB directory. DOS was in the root directory. The batch file changed directories and set up a path. Then the Basic program RXXP05 was called. This was the initial log-on menu program. Logging on involved supplying a password. The password was encrypted and written to a file called RXXFPWD.FEX. As other programs in the system were executed, they checked to see that a proper password was set before proceeding. When the operator exited the system, the password file was emptied.

The RXXP05 program wrote information to a batch file called RXXB00. After completion of the RXXP05 program, execution was returned to the STRTACDB batch file which then called the RXXB00 batch file. Other programs continued this chain, modifying the RXXB00 file and another batch file called RXXB10 until the system was exited through the RXXB05 program. Some programs such as the sort utility had their own special batch file. These are explained later.

3.5 OTHER PROGRAMS

Two other programs which resided in the scanner XT are COSORT and SCANUTIL. These programs were acquired from outside vendors. COSORT was used to sort the data before loading into the FOCUS data base although some of these files were sorted in the Data Base XT system. SCANUTIL was a scanner utility program provided by NCS and used primarily in development to examine the scanner output records.

3.6 COSORT

COSORT Version 4.1 was a group of programs obtained from Information Resources in New York. These programs are used to sort large data bases efficiently. A detailed explanation will not be given in this documentation since there is a COSORT Version 4.1 manual available. Specific details of how it was used in the scanner XT system will be given.

In the scanner XT the COSORT programs were used to sort the provider and patient registration data which had been scanned and placed in an ASCII output file. Three items were needed:

1. COSORT.COR
2. RECOUP.EXE
3. A batch file with sort parameters.

The first two items are the sort programs from Information Resources. The third item is a batch file customized to sort the user's data. Since the Provider and Patient files were to be sorted in the scanner XT system, a batch file for each was needed. They are

RACPRSRT.BAT	(Provider batch file)
RACPTSRT.BAT	(Patient batch file)

The contents of each batch file are given below:

	<u>RACPRSRT.BAT</u>	<u>RACPTSRT.BAT</u>
1.	sort	sort
2.	ccc	ccc
3.	0	0
4.	1	1
5.	provider.dat	patient.dat
6.	1	1
7.	ascending	ascending
8.	fixed	fixed
9.	1	1
10.	17	20

11.	alpha	alpha
12.	none	none
13.	no	no
14.	file	file
15.	provider.dat	patient.dat

The line numbers above are included to aid in explaining the parameters and are not part of the file.

Line 1. Type of operation.
 Line 2. Name of work disk (repeated three times).
 Line 3. Record length (0 for variable length).
 Line 4. Number of input files.
 Line 5. Input file name.
 Line 6. Number of sort keys.
 Lines 7-13 Sort key parameters, could be repeated if more than one sort key is required.
 Line 7. Sort direction.
 Line 8. Field locator.
 Line 9. Position of field.
 Line 10. Length of field.
 Line 11. Character type.
 Line 12. Alignment.
 Line 13. Case change.
 Line 14. Device to send output records.
 Line 15. Name of output file.

The COSORT program was called from the Provider BASIC program RACP300 by setting up a batch file with the line: 'RECOUP RACPRSRT.BAT' in it. After completion the RACP300 program returned to the batch file which in turn executed the RECOUP program using the parameters from the RACPRSRT.BAT file. The Patient BASIC program called the sort program in a similar fashion.

3.6.1 WARNING

WARNING! The sort program can get in a loop when it runs out of disk space. No indication is given to the operator except the program continues to run much longer than it should. The operator has a couple of options available when this happens. As in most cases when the unexpected occurs, no explicit set of rules applies. However, the operator could interrupt the program and free up additional disk space before continuing the program or the operator could scan fewer forms so that the file to be sorted is smaller.

3.7 SCANUTIL PROGRAM

The SCANUTIL program was provided by NCS to be used in development of the scanner system. SCANUTIL is a

utility program that can drive the scanner and capture the scanner output records as each form is scanned.

The program SCANUTIL.EXE is a stand-alone program and is executed by typing 'SCANUTIL' at the DOS prompt. It is menu driven and for the most part self explanatory.

3.7.1 MAIN MENU

The first menu to appear is the MAIN menu. Its format is as follows:

COPYRIGHTED REL 1.1
NATIONAL COMPUTER SYSTEMS INC.

SCAN UTILITY MAIN MENU

- 0 - EXIT FROM SCAN UTILITY
- 1 - SCAN TO DISK
- 2 - SCAN TO SCREEN
- 3 - SCAN TO PRINTER
- 4 - SET UP CONFIGURATION

SELECT OPTION (0-4):

3.7.2 CONFIGURATION MENU

The configuration had to be set up each time the program was executed. The configuration menu and the correct parameters for the ACDB Scanner XT system are as follows:

SCAN UTILITY
SYSTEM CONFIGURATION

- 0 - EXIT TO MAIN MENU
- 1 - SCANNING MODE:....OPERATOR ACTIVATED
- 2 - DISPLAY/PRINT FORMAT:.....TEXT
- 3 - SCANNER SERIES: 7000 OR 9000
- 4 - BAUD RATE:.....9600
- 5 - PARITY:.....NONE
- 6 - DATA BITS:.....8
- 7 - STOP BITS:.....1
- 8 - SERIAL PORT:.....2

SELECT OPTION (0-8):

Items 4-8 above had to be set each time the program was loaded. By selecting an option, the operator would see another menu with choices available for that option. The options used in the ACDB system are BAUD RATE = 9600, PARITY = NONE, DATA BITS = 8, STOP BITS = 1 and SERIAL PORT = 2.

3.7.3 DISPLAY RECORD

From the main menu the output could be displayed on the screen, printed, or written to a file. Generally the records were printed because close scrutiny of the record was required as well as a hard copy in case of future problems. A sample of a scanner output record printed by the SCANUTIL program for the Allergy Form (410) is shown below:

SCAN UTILITY		SOURCE SCANNER	
SHEET RECORD DISPLAY		RECORD NUMBER: 1 RECORD LENGTH: 276	
FORMAT: TEXT			
0	1	2	3
1.....0.....0.....0.....0.....0.....0			
4100010001101	00013936	0402T123498700123401	12A00
1	88888133333	1	
1 1		1	271
	1 1		

The sample is truncated to fit on this page. It normally extends to 80 columns. The scanner has converted some data and performed some editing but now sends it to the XT for further editing and conversion. Notice the form number, 410, in the first three characters. The provider ID, T1234, can be seen in

positions 39-43. Position details are provided in section 2.4.3.

3.8 REFERENCES

BASIC Compiler. (1982) Boca Raton, FL: International Business Corporation.

COSORT Program Manual Version 4.1. (1984) Manhasset, NY: Information Resources, Inc.

Disk Operating System Version 3.1. (1985) Boca Raton, FL: International Business Corporation.

4.0 DATA BASE IBM XT SYSTEM

4.1 OVERVIEW OF DATA BASE IBM XT SYSTEM

An IBM XT system was used to provide storage and reporting capabilities through a fourth generation data base management package called FOCUS. Initially FOCUS resided with the Scanner system, but, because of the large amounts of time required to load, dump and process the data, two changes were made. The external disk drives and cartridges were increased from 10M to 20M bytes and a separate IBM XT was acquired so that the data base system could be run independently. The data base system allowed reports and other procedures to be run while new forms were being scanned on a separate system.

4.2 EQUIPMENT

The data base management system consisted of an IBM XT with a 10M byte disk drive, a color monitor and 256K of memory. A memory board supplied with FOCUS contained 512K of memory, bringing the total memory to 768K. Early versions of FOCUS took advantage of the memory above 640K. An Iomega Bernoulli 20M dual drive was used for data storage. A Case-Rixon model PC212A internal modem operating at 1200 baud was used for electronic mail, however, it was installed in either the scanner or data system depending which was more convenient to a phone line. A Genicom Model 3014 printer with a wide carriage was installed.

The FOCUS memory boards, particularly a piggy-back memory card containing 256K bytes, caused problems after being used about a year. The clock battery also seemed to quit after a year of use. When the 20M byte Iomega disk drives were installed, the piggy-back card on some FOCUS boards had to be removed before the Iomega drive would work. The FOCUS board was essential, however, since FOCUS versions 1.5 and earlier versions had a copy protection scheme incorporated within the board. After FOCUS version 2.0 was installed, any memory board would work, and, in some cases, AST SIXPAK boards with 384K bytes of memory were used as replacements.

In addition to the computer system, a SOLA uninterruptable power supply was used at most of the sites unless the equipment was placed in a computer room with a backup power supply.

4.3 FUNCTIONAL DESCRIPTION

The forms were scanned and edited by the scanner system. The data was written in flat (ASCII) files on the Iomega cartridges. These cartridges were then taken to the data base system where the data was loaded into the hierarchical data base called FOCUS.

There were six separate data bases that could be joined for reporting. They were a Provider file, a Patient file, a Clinic description file, a Procedure description file, a Diagnosis description file and a Visit file. The Visit file contained the data from the encounter forms and was cumulated monthly. At the end of each month reports were printed, and the data then dumped onto cartridges and sent to Fort Detrick to be loaded on a mainframe computer.

A login procedure was necessary to prevent unauthorized access to the system. After logging on, the operator was guided through the available options by a menu system.

4.4 SOFTWARE

4.4.1 SYSTEM SOFTWARE

FOCUS version 1.0 was first installed as the data base management language. FOCUS 1.5 soon came along and the system was upgraded. Two additional upgrades of FOCUS 1.5 were installed and used until FOCUS 2.0 was installed. Early versions of FOCUS had a number of problems which caused a great deal of difficulty in development efforts as well as in providing technical support to the sites. Early documentation was also very poor. New documentation with FOCUS 1.5 was a significant improvement, and each new version of software fixed a number of problems present in the older version. Most of the bugs causing problems in the ACDB system were removed by the time FOCUS 2.0 was installed. However, at that time the system was using DOS 3.0, and FOCUS 2.0 seemed to lockup the system rather frequently. The computer had to be turned off and then on again to clear the lockup. FOCUS personnel at Information Builders, Inc. (IBI) could not resolve the problem; upgrading to DOS 3.1 reduced the lockup problem considerably. Programming details of how to use FOCUS will not be discussed in this manual. See P/C FOCUS Users manual release 2.0. IBI has since released FOCUS version 3.0 with a new manual.

Portions of the menu system were written in the

BASIC language under the IBM Basic Compiler version 1.0. Other software included COSORT Version 4.0.

4.5 FOCUS PROGRAMS

4.5.1 OVERVIEW

The FOCUS programs (or procedures as the FOCUS documentation calls them) developed for the ACDB Study could be grouped into several categories. There were the load programs, the reporting programs, the correction programs, the dump programs and other utility programs.

The load procedures read the flat files written by the scanner system and loaded them into the FOCUS data base. FOCUS has several types of procedural languages. The load procedures were written as Modify Requests.

The reporting procedures, called Table Requests by FOCUS, print or display on the computer screen various formatted reports. A number of these Table Requests were developed as standard reports and sent to each site as part of the menu system. They could be called up and run if desired. Other Table Requests were written by site personnel. These were customized reports for their particular sites and, in some cases, were shared among the sites.

Correction procedures are a combination of Table Requests and Modify Procedures used to correct errors found in the data base. FOCUS has a package called SCAN that can be used to correct a FOCUS data base; it is complex, and it can introduce more errors or destroy portions of the data base when used by inexperienced personnel.

Dump procedures are Table Requests that send the output to a file rather than a printer. These files are written on the Iomega cartridges and sent to Fort Detrick for inclusion on the mainframe data base also in FOCUS.

The FOCUS language used outside of Table Requests and Modify Requests to control administrative tasks is called Dialogue Manager. Portions of the menu system were written in Dialogue Manager. These procedures, called the Load, Dump, Reporting and Correcting procedures, returned to the BASIC menu system after execution.

4.5.2 PROFILE

A file called PROFILE.FEX may reside in a directory in which FOCUS is invoked. This file is similar to the AUTOEXEC.BAT file in DOS in that it executed automatically each time FOCUS is called. The Basic programs that began an ACDB session had the capability to write a PROFILE file for specific circumstances. Upon exiting from the system, the Basic program left a generic PROFILE in the directory. The generic form of PROFILE.FEX is in Appendix D.

4.6 MASTER FILES

The structure of a FOCUS data base had to be described in a master file. The master file had a MAS extension in the PC and contained the hierarchical structure as well as the field names and their attributes. The master file described the attributes of a data file with the same name, but one having an extension of FOC.

The data collected from all the encounter forms was stored in a file called VISIT.FOC. A listing of the VISIT.MAS file is provided in Section 4.6.1 to show the structure of the file and all field names with their attributes. The first line shows the file name. The second line, and all other lines that begin with SEGNAME describe the segment names and their relation to other segments. The second line has no parent and is, therefore, the root segment. A pictorial description of the segments for the VISIT file is shown in Section 4.6.1.1. The first line in each box is the segment name and the other line is the first field name in that segment.

The lines following the SEGNAME lines in the master file description give the field names and their attributes for each segment. The first name following 'FIELD=' is the full field name and is used as a default report heading for that field. The second name is an alias which primarily is an abbreviated name used in writing FOCUS procedures (programs). The next item in the field line is the field type. Generally the types are either integer or alphabetical and are indicated in the master file as an I or A followed by the size of the field. Thus the third line has a field name called CLINIC, an alias called CL_UCA, and is an alphabetic field having a length of four characters. The only exception to field type is in the fifth line for VISIT_DATE. Here the size is followed by YMD indicating the date is stored year, followed by month, followed by day.

4.6.1 VISIT MASTER FILE

```

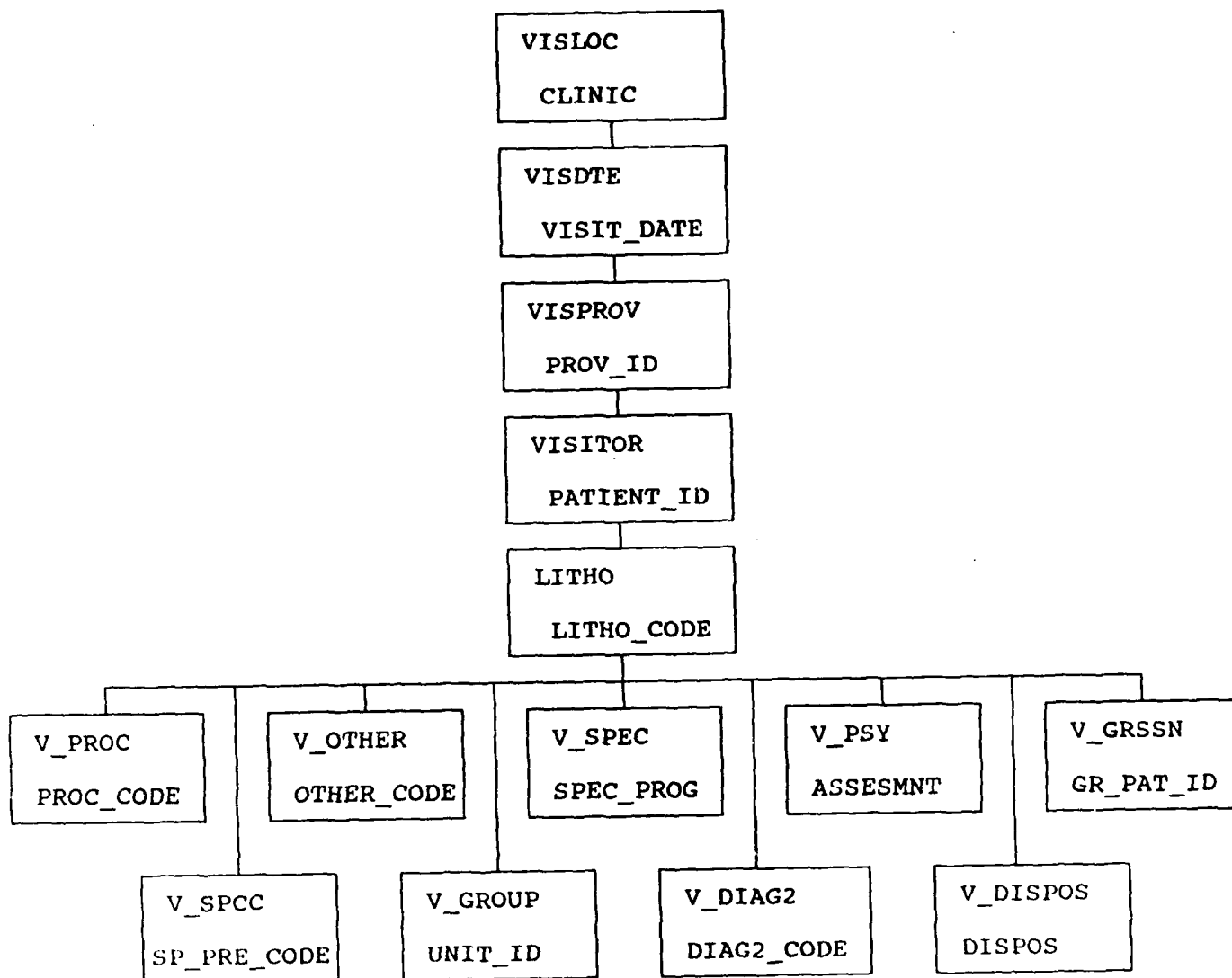
FILE=VISIT,SUFFIX=FOC,$
SEGNAME=VISLOC,SEGTYPE=S1,$
  FIELD=CLINIC,CL_UCA,A4,$
SEGNAME=VISDTE,PARENT=VISLOC,SEGTYPE=S1,$
  FIELD=VISIT_DATE,VDATE,I6YMD,$
SEGNAME=VISPROV,PARENT=VISDTE,SEGTYPE=S1,$
  FIELD=PROV_ID,PROV1,A5,$
SEGNAME=VISITOR,PARENT=VISPROV,SEGTYPE=S1,$
  FIELD=PATIENT_ID,PTID,A11,$
SEGNAME=LITHO,PARENT=VISITOR,SEGTYPE=S1,$
  FIELD=LITHO_CODE,LITHO,I8,$
  FIELD=FORM_NUM,FNUM,A2,$
  FIELD=VISIT_CNT,VCNT,I1,$
  FIELD=PROV1_TIME,TIME1,I3,$
  FIELD=PROVIDER_2,PROV2,A5,$
  FIELD=PROV2_TIME,TIME2,I3,$
  FIELD=PROV2_RESON,PROV2RES,A1,$
  FIELD=APPT_STATUS,APPSTAT,A1,$
  FIELD=PLACE_OF_VIS,PLACE,A1,$
  FIELD=INP_REFERAL,REFERAL,A4,$
  FIELD=JOB_RELATED,JOBREL,A1,$
  FIELD=MIL_DUTY,DUTY,A1,$
  FIELD=MIL_QTR,QTR,A1,$
  FIELD=MIL_PROF,PROF,A1,$
  FIELD=NOT_AVAIL,NAVAIL,A1,$
  FIELD=ADMITTED,ADMIT,A1,$
  FIELD=INFIELD,ILLN,A1,$
  FIELD=INJURIES,INJ,A1,$
  FIELD=PURP_VIS,PURP,A1,$
  FIELD=RO_F_UP,RULE,A1,$
  FIELD=DIAG1_CODE,DX1CODE,A5,$
SEGNAME=V_PROC,PARENT=LITHO,SEGTYPE=S1,$
  FIELD=PROC_CODE,PCODE,A5,$
  FIELD=PROV_CD,PCODE,A1,$
SEGNAME=V_SPCC,PARENT=LITHO,SEGTYPE=S1,$
  FIELD=SP_PRE_CODE,SPCC,A1,$
SEGNAME=V_OTHER,PARENT=LITHO,SEGTYPE=S1,$
  FIELD=OTHER_CODE,OTHER,A5,$
SEGNAME=V_GROUP,PARENT=LITHO,SEGTYPE=S1,$
  FIELD=UNIT_ID,UIC,A6,$
  FIELD=TIME_PR1,TIMPR1,I3,$
  FIELD=TIME_PR2,TIMPR2,I3,$
  FIELD=TIME_TR1,TIMTR1,I3,$
  FIELD=TIME_TR2,TIMTR2,I3,$
  FIELD=NO_ACT_DUTY,MACT,I3,$
  FIELD=NO_OTH_DUTY,OACT,I3,$
  FIELD=NO_RET_MIL,RMIL,I3,$
  FIELD=NO_DEPEND,DEPD,I3,$
  FIELD=NO_CIVIL,CIVIL,I3,$
  FIELD=CONT_SHEET,CSHEET,A1,$

```

SEGNAME=V_SPEC,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=SPEC_PROG,SPROG,A1,\$
SEGNAME=V_DIAG2,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=DIAG2_CODE,DX2CODE,A5,\$
SEGNAME=V_PSY,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=ASSESMNT,ASSESMNT,A1,\$
SEGNAME=V_DISPOS,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=DISPOS,DISP,A2,\$
SEGNAME=V_GRSSN,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=GR_PAT_ID,GPATID,A11,\$
END
DBA=LUGNUT,\$
USER=NUTMEG,ACCESS=RW,\$

4.6.1.1 PICTURE OF VISIT FILE

PICTURE DESCRIPTION OF FOCUS DATA BASE



4.6.2 PATIENT MASTER FILE

```
FILE=PATIENT,SUFFIX=FOC
SEGNAME=PTSEG,SEGTYPE=S1
  FIELD=PATIENT_ID,PTID,A11,FIELDTYPE=I,$
  FIELD=PT_DOB,PTDOB,I6YMD,$
  FIELD=PT_CATEGORY,CATEG,A3,$
  FIELD=PT_LITHO,PTLITHO,A8,$
  FIELD=PT_REGDATE,REGDATE,I6YMD,$
  FIELD=PT_GENDER,SEX,A1,$
  FIELD=PT_RACE,RACE,A1,$
  FIELD=PT_PREFIX,PREFIX,A2,$
  FIELD=PT_PAY_GRADE,PAYGRADE,A2,$
  FIELD=PT_JOB_CODE,JOBCODE,A4,$
  FIELD=PT_LOCATION,PTLOC,A6,$
  FIELD=PT_TRN_TDY,TRAINEE,A1,$
  FIELD=PT_VAELIG,VA,A1,$
  FIELD=PT_HCI,HCI,A1,$
  FIELD=PT_ZIP_CODE,ZIP,A5,$
  FIELD=PT_DUAL_SSN,PTDUALSSN,A9,$
  FIELD=PT_FOREIGN,PTFORGN,A3,$
END
DBA=LUGNUT,$
USER=NUTMEG, ACCESS=RW, $
```

4.6.3 PROVIDER MASTER FILE

```
FILE=PROVIDER,SUFFIX=FOC
SEGNAME=PROVIDER,SEGTYPE=S1
  FIELD=PROV_ID,PROVID,A5,FIELDTYPE=I,$
  FIELD=PROV_DATE,PROVDATE,I6YMD,$
  FIELD=PROV_LST_NME,PRLNME,A16,$
  FIELD=PROV_FST_NME,PRFNME,A13,$
  FIELD=PROV_INIT,PRINIT,A1,$
  FIELD=PROV_STAT,PRSTAT,A1,$
  FIELD=PROV_CATEG,PRCATEG,A2,$
  FIELD=PROV_PRPOS,PRPOS,A2,$
  FIELD=PROV_PAYGR,PRPAYGR,A2,$
  FIELD=PROV_JOBCODE,PRJCODE,A7,$
  FIELD=PROV_SSN,PRSSN,A9,$
  FIELD=PROV_LITHO,PRLTH,A8,$
END
DBA=LUGNUT,$
USER=NUTMEG, ACCESS=RW, $
```

4.6.4 CLINIC MASTER FILE

FILENAME=CLINIC,SUFFIX=FOC

SEGNAME=CLINIC,SEGTYPE=S1

FIELDNAME=CLINIC, ALIAS=CLUCA,

FORMAT=A4, FIELDTYPE=I, \$

FIELDNAME=CL_TITLE, ALIAS=302CLTITLE, FORMAT=A20, \$

FIELDNAME=CL_302CODE, ALIAS=302CLCODE, FORMAT=A2, \$

FIELDNAME=CL_302LINE, ALIAS=302CLLINE, FORMAT=I3, \$

END

DBA=LUGNUT, \$

USER=NUTMEG, ACCESS=RW, \$

4.6.5 DIAGNOSIS MASTER FILE

FILENAME=DIAGNOS,SUFFIX=FOC

SEGNAME=DIAGNOS,SEGTYPE=S1

FIELDNAME=DIAG_CODE, ALIAS=DGNCD, FORMAT=A5, FIELDTYPE=I, \$

FIELDNAME=DIAG_DESCR, ALIAS=DGNDSC, FORMAT=A50, \$

END

DBA=LUGNUT, \$

USER=NUTMEG, ACCESS=RW, \$

4.6.6 PROCEDURE MASTER FILE

FILENAME=PROCEDUR,SUFFIX=FOC

SEGNAME=PROCEDUR,SEGTYPE=S1

FIELDNAME=PROC_CODE, ALIAS=PRCCD, FORMAT=A5, FIELDTYPE=I, \$

FIELDNAME=PROC_DESCR, ALIAS=PRCDSC, FORMAT=A67, \$

END

DBA=LUGNUT, \$

USER=NUTMEG, ACCESS=RW, \$

4.6.7 OTHER CODE MASTER FILE

FILENAME=OTHER,SUFFIX=FOC

SEGNAME=OTHERCOD,SEGTYPE=S1

FIELDNAME=OTHER_CODE, ALIAS=OTHC, FORMAT=A5, FIELDTYPE=I, \$

FIELDNAME=OTHER_DESCR, ALIAS=OTHDC, FORMAT=A67, \$

END

DBA=LUGNUT, \$

USER=NUTMEG, ACCESS=RW, \$

4.7 FIELD DESCRIPTIONS

4.7.1 VISIT FIELD DESCRIPTIONS

The following list describes all the fields in the VISIT file providing additional information such as a description of acceptable input values, ranges, and names as given on the encounter form:

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
CLINIC	CL_UCA	Clinic code
VISIT_DATE	VISIT_DAT	Visit date
PROV_ID	PROV1	Provider 1 ID
PATIENT_ID	PTID	SSN and FMP
LITHO_CODE	LITHO	Form serial number
FORM_NUM	FNUM	Form type
VISIT_CNT	VCNT	Visit count (0-9)
PROV1_TIME	TIME1	Provider 1 time (minutes)
PROVIDER_2	PROV2	Provider 2 ID
PROV2_TIME	TIME2	Provider 2 time (minutes)
PROV2_RESON	PROV2_RES	Reason for Prov 2 <ol style="list-style-type: none"> 1 Teaching 2 Consultation 3 Procedure 4 Other 1 5 Other 2 6 Other 3 7 Co-therapist
APPT_STATUS	APPSTAT	Appointment Status <ol style="list-style-type: none"> 1 Scheduled (Also ER sched foll-up) 2 Unscheduled 3 Emergency 4 Walk in 5 Sick call
PLACE_OF_VIS	PLACE	Place of visit <ol style="list-style-type: none"> 1 Clinic 2 Ward 3 Telephone 4 Home 5 Other 1 6 Other 2 7 Other 3 8 ER Seen last 48 hr
INP_REFERAL	REFERAL	Referral Clinic code
JOB_RELATED	JOBREL	Job related (Y/N)
		OCC HEALTH <ol style="list-style-type: none"> N Not job related 1 Fatal case 2 Lost time case 3 No lost time case

MIL_DUTY	DUTY	Military Duty 1 Duty 2 Limited Duty 3 TMC 4 Self Care Protocol
MIL_QTR	QTR	Military Quarters 1 24 hours 2 48 hours 3 72 hours
MIL_PROF	PROF	Military Profile 1 1-3 days 2 4-7 days 3 8-14 days 4 > 14 days
NOT_AVAIL	NAVAIL	No Data in this field See SPEC PRE CLINIC
ADMITTED INFIELD	ADMIT ILLN	Admitted (Y/N) For BAS: Field (Y/N) For Occ He: Illness N New R Revisit
INJURIES	INJ	Injuries BAS/TMC 1 PT 2 Unit sport 3 MOS/duty 4 Field duty 5 Motor veh accident 6 Airborne Ex. OCC HEALTH N New R Revisit
PURP_VIS	PURP	Purpose of visit (Occ He) A Preplacement/preass B Periodic C Termination D Baseline hlth assess E Pregnancy surv F Chronic dis surv G Voluntary hlth scr H Follow-up I Fitness for duty J Disab retirement K Illness abs mon L Med eval (claimant) M Med eval of empl
RO_F_UP	RULE	Rule out/Follow-up 1 Rule out 2 Follow-up
DIAG1_CODE PROC_CODE	DX1CODE PCODE	Primary Dx code Procedure code

PROV_CD	PRCODE	Provider giving proc 1 Prov 1 2 Prov 2 3 Out
SP_PRE_CODE	SPCC	Specific preassigned codes (1-9) Not Available M Medical records L Lab results X Xrays
OTHER_CODE UNIT_ID	OTHER UIC	Other codes (Incl M codes) Unit ID code Time spent with group coded as prov1 and prov2 time
TIME_PR1	TIMPR1	Time spent prep prov 1
TIME_PR2	TIMPR2	Time spent prep prov 2
TIME_TR1	TIMTR1	Time spent travel prov 1
TIME_TR2	TIMTR2	Time spent travel prov 2
NO_ACT_DUTY	MACT	Number Army Active duty
NO_OTH_DUTY,	OACT	Number Other Active duty
NO_RET_MIL	RMIL	Number Retired Military
NO_DEPEND	DEPD	Number dependents
NO_CIVIL	CIVIL	Number civilians
CONT_SHEET	CSHEET	Continuation sheet
SPEC_PROG	SPROG	Special programs 1 FAP 2 ADAPCP 3 PRP 4 Adoption 5 Other 1 6 Other 2 7 Other 3
DIAG2_CODE ASSESMNT	DX2CODE ASSESMNT	Secondary Dx code Psychometric Assessment 1 Personality 2 Intellectual 3 Neuropsychological 4 Other
DISPOS GR_PAT_ID	DISP GPATID	Dispositions (1-13) Group patient ssN & fmp

4.7.2 PATIENT FIELD DESCRIPTIONS

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
PATIENT_ID	PTID	Social Security number followed by FMP (11 digits)
PT_DOB	PTDOB	Date of Birth (yyymmdd)
PT_CATEGORY	CATEG	Category
		A10 Army Active Duty
		A21 Army Reserve Tng
		A23 Army Reserve IDT/AT/ADT
		A24 Army NG Tng
		A26 Army NG IDT/AT/ADT
		A30 Army Retired
		A50 Army Depn Active Duty
		A60 Army Depn Retired/dec.
		A70 Army USMA Cadet
		A80 Army ROTC Cadet/Applic.
		C10 CG Active Duty
		C20 CG Reserve/NG
		C30 CG Retired
		C50 CG Depn Active Duty
		C60 CG Depn Retired/dec.
		C70 CG Academy Cadet
		F10 AF Active Duty
		F20 AF Reserve/NG
		F30 AF Retired
		F50 AF Depn Active Duty
		F60 AF Depn Retired/dec.
		F70 AF Academy Cadet
		F80 AF ROTC Cadet/Applic.
		H26 Civilian
		N10 Navy Active Duty
		N20 Navy Reserve/NG
		N30 Navy Retired
		N50 Navy Depn Active Duty
		N60 Navy Depn Retired/dec.
		N70 Navy Academy Cadet
		N80 Navy ROTC Cadet/Applic.
		M10 Marine Active Duty
		M20 Marine Reserve/NG
		M30 Marine Retired
		M50 Marine Depn Active Duty
		M60 Marine Depn Retired/dec.
		P10 USPHS Active Duty
		P20 USPHS Reserve
		P30 USPHS Retired
		P50 USPHS Depn Active Duty
		P60 USPHS Depn Retired/dec.
		OTH
		O10
		O20

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
		O30
		O50
		O60
		H10
		H20
		H30
		H38
		H40
		H50
		J10
		J20
		J30
		K10
		K20
		K30
		K40
		K42
		K43
		K50
		K51
		K60
		K70
		Q10
		R10
		S10
		S20
		S23
		S24
		S30
		S32
		S40
		S50
		S60
		X10
		X20
		X30
		X40
		X45
		X52
		X60
		X70
PT_LITHO	PTLITHO	Form serial number (8 digits)
PT_REGDATE	REGDATE	Registration date (yyymmdd)
PT_GENDER	SEX	Sex (M,F)

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
PT_RACE	RACE	Race
		1 Am. Indian/Alaskan Native
		2 Asian/Pacific Islander
		3 Black/Not Hispanic Origin
		4 Black/Hispanic Origin
		5 White/Not Hispanic Origin
		6 White/Hispanic Origin
		7 Unknown
PT_PREFIX	PREFIX	Position
		E Enlisted
		O Officer
		W Warrant Officer
		GS Civilian
		GM Civilian
		WG Civilian
		WL Civilian
		WS Civilian
		NF NAF
		CT Contract
		OT Other
PT_PAY_GRADE	PAYGRADE	Pay grade (01-16)
PT_JOB_CODE	JOB CODE	Duty MOS/SSI or Civilian Occ. Series
PT_LOCATION	PTLOC	Unit ID Code or Building No.
PT_TRN_TDY	TRAINEE	Trainee or on TDY (Y,N)
PT_VAELIG	VA	Eligible for VA benefits (Y,N)
PT_HCI	HCI	Health Care Insurance (Y,N)
PT_ZIP_CODE	ZIP	Zip code
PT_DUAL_SSN	PTDUALSSN	Civilian SSN
PT_FOREIGN	PTFORGN	Foreign National Country Code

4.7.3 PROVIDER FIELD DESCRIPTIONS

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
PROV_ID	PROVID	First initial of last name plus last four digits of SSN
PROV_DATE	PROVDATE	Registration date
PROV_LST_NME	PRLNME	Last name
PROV_FST_NME	PRFNME	First name
PROV_INIT	PRINIT	Middle initial
PROV_STAT	PRSTAT	Status
		1 Permanent staff at MTF
		2 All others

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
PROV_CATEG	PRCATEG	Category (2 digits) First digit - Branch 1 Army 2 Air Force 3 Navy 4 Marines 5 Coast Guard 6 PHS 7 Other Second digit - Category 1 Officer 2 Warrant Officer 3 Enlisted 4 Civilian (DAC) 5 Contract 6 Consultant 7 Other
PROV_PRPOS	PRPOS	Position 1 CDR/DCCS 2 Chief Dept./Svc. 3 Fellow 4 Resident 5 Intern 6 Nurse Pract. 7 Nurse Clin. 8 NCOIC 9 Student 10 All Others
PROV_PAYGR	PRPAYGR	Pay grade (2 char.) First Char. - Prefix E Enlisted O Officer W Warrant Officer Second Char. - pay grade (1-9)
PROV_JOBCODE	PRJCODE	Job code (7 char) Army Char. 1-3 - MOS Char. 4 - SSI Char. 5,6 - ASI Char. 7 - blank Air Force Char. 1-7 - AFSC Navy Char. 1-6 - NOBC/NEC Char. 7 - blank Other Branches - 7 blanks
PROV_SSN	PRSSN	Provider SSN
PROV_LITHO	PRLTH	Form serial number (8 digits)

4.7.4 CLINIC FIELD DESCRIPTIONS

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
CLINIC	CLUCA	UCA Code for Clinic
CL_TITLE	302CLTITLE	Clinic description
CL_302CODE		Not used
CL_302LINE		Not used

4.7.5 DIAGNOSIS FIELD DESCRIPTIONS

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
DIAG_CODE	DGNCD	Diagnosis code
DIAG_DESCR	DGNDSC	Diagnosis description

4.7.6 PROCEDURE FIELD DESCRIPTIONS

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
PROC_CODE	PRCCD	Procedure code
PROC_DESCR	PRCDSC	Procedure description

4.7.7 OTHER FIELD DESCRIPTIONS

<u>FIELD</u>	<u>ALIAS</u>	<u>DESCRIPTION</u>
OTHER_CODE	OTHCD	Other code
OTHER_DESCR	OTHDSC	Other code description

4.8 LOADING

Data was loaded into the FOCUS data base from ASCII files produced by the scanner system or manually prepared. Listings of the Load procedures are in Appendix D. They are: LOADCLIN.FEX, LOADDIAG.FEX, LOADPROC.FEX, LOADOTH.FEX, RXXF200.FEX, RXXF300.FEX, and RXXF400.FEX. The first four programs loaded descriptions for the clinic, diagnosis, procedure, and other codes. The last three programs loaded the patient, provider, and visit data respectively. One other load program, DEERS.FEX, was written later in the study to load the DEERS data.

4.9 REPORTING

Report programs were written early in the study to provide the sites with an immediate use for the data.

Since the precise requirements of each site were not known, general reporting programs were designed. Many sites used these programs, but as the site operators became experienced in FOCUS, customized programs were written and used.

There were 12 report programs initially provided, however, two were never completed. The programs are

CLINIC1.FEX	MILT3.FEX
CLINIC2.FEX	PROV1.FEX
CLINIC3.FEX	PROV2.FEX
CLINIC4.FEX	PROV3.FEX
MILT1.FEX	PROV4.FEX
MILT2.FEX	PROV5.FEX

Listings of the above programs are in Appendix D.

4.10 CORRECTING

It became obvious early in the study that errors in the data base caused various reporting anomalies. The ability to correct the errors through FOCUS was limited. An available SCAN utility could cause the data base to become corrupt beyond repair. It could also cause other problems such as duplicate keys. Only an experienced programmer should attempt to use SCAN and then only under emergency conditions.

Another problem lay in the correction of key fields. FOCUS did not allow these fields to be corrected through the modify routines which could be used for correcting data. Thus an attempt was made to write some programs that could correct the data to include key fields and that would be safe to use. Three programs were written to correct the Clinic field, Provider field and the Patient ID field. The programs were RXXFCCL.FEX, RXXFCPR.FEX and RXXFCPA.FEX. Supporting programs for the above are RXXCORL.FEX, RXXCCLD.FEX, RXXCPAD.FEX, and RXXCPRD. Listings of these programs are in Appendix D.

The correction programs provided a menu system through which errors in certain fields could be changed. The affected segments were then correctly written out to a temporary file. The temporary file was then read back into the data base. Finally, the incorrect records were deleted. Although the method worked, it was time consuming, particularly at end of month processing when the data base was large. Typically a single correction (which may affect one or hundreds of segments) took over an hour.

4.11 DUMPING

Each month the data collected at the sites had to be sent to Ft. Detrick for inclusion in the total data base. Since the data could not be uploaded in the FOCUS structure, an ASCII file was created. As the data was medically sensitive, identifying fields were encrypted during the dump. Thus the data was protected the instant it left the site. Dump programs for the visit, patient, and provider files are in Appendix D. They are RXXFDVI.FEX, RXXFDPA.FEX, and RXXFDPR.FEX respectively.

5.0 SPECIAL TOPICS

Some topics span more than one system and are therefore inconvenient to classify. A special topic section is devoted to these cases. One such topic is Patient registration, the problems associated with it, and attempts to bolster registrations with the SIDPERS and the DEERS data.

5.1 PATIENT REGISTRATION

A form for patient registration was designed, and an attempt to register all out-patients was made. However, after a few months patient registrations fell behind at all sites, and the Patient file represented about a third of the patients being seen. In an attempt to bolster the patient registrations, the study team drew upon data from two outside sources - SIDPERS and DEERS.

The data from SIDPERS and DEERS were sent directly to Ft Detrick. At Ft. Detrick the data had to be converted to the format design for the ACDB Study. Then a data set was sent to the site for inclusion in their Patient file, and a copy was added to the cumulative Patient file on the mainframe computer.

Addition of the SIDPERS and DEERS data brought the percentage of registrations to over 65% in some cases; however, not all fields on the registration forms were available in the outside data. Another difficulty was that the Patient file began to grow beyond the storage capabilities of the ACDB computer systems. The problem was that many individuals that never made a hospital visit were being added to the data base each month. Although this problem was discussed, no good solution was found for deleting these unwanted individuals from the data base. This is an area warranting further review, research and testing.

5.2 SIDPERS

SIDPERS data was sent by tape monthly from Ft. Polk, Ft. Jackson, Ft. Campbell and Ft. Bragg since these sites had a large turnover in trainees. The data represented new personnel arriving each month. More information was collected on the registration forms than was available from SIDPERS. Also, some of the data was in a different format. The data from SIDPERS had to be transformed before being added to the patient data base.

5.2.1 SIDPERS TAPE FORMAT

The SIDPERS tapes were requested to be sent unlabeled and in EBCDIC (IBM compatible). Each record was 30 characters in length and contained the following fields:

<u>Column</u>	<u>Description</u>
1-9	Social Security Number
10	MPC
11	Grade
12	Sex
13-17	Duty MOS
18-22	UPC
18-22	DOB (yyymmdd)
29	Race
30	Blank

5.2.2 SIDPERS CONVERSION

The data from the SIDPERS tapes had to be converted to conform to the ACDB. Conversion was made using a BASIC program called CONVERT.EXE on the PC for files being sent back to the sites. A mainframe program, SIDCONV, written in SAS performed the conversion for the mainframe Patient file. SIDCONV also added the site ID to each record and encrypted the Social Security Number.

The conversions performed are as follows:

GRADE

Enlisted (MPC = E)

<u>GRADE</u>	<u>CONVERT TO</u>
M	04
N	05
O	06
P	07
R	08
X	01
Y	02
1-9	01-09

Officer (MPC = O)

<u>GRADE</u>	<u>CONVERT TO</u>
A	08
B	06
C	05
D	04
E	03

F	02
G	01
5	03
6	02
7	01

Warrant Officer (MPC = W)

<u>GRADE</u>	<u>CONVERT TO</u>
U	04
V	03
W	02
X	01

RACE

<u>RACE</u>	<u>CONVERT TO</u>
blank	7
C	5
M	2
N	4
R	1
X	7

JOB CODE

If MPC = E then JOB CODE is 0 plus left two positions of Duty MOS plus a W.

If MPC = O then JOB CODE is 0 plus left two positions of Duty MOS plus a W.

If MPC = W then JOB CODE is left two positions of Duty MOS plus a blank plus a W.

OTHER ADDED FIELDS

PROGID = 200	Program ID
FMP = 20	Family member prefix
CAT = A10	Category
ZIP = 28307	Zip Code for Ft. Bragg
= 42223	Zip Code for Ft. Campbell
= 29207	Zip Code for Ft. Jackson
= 71459	Zip code for Ft. Polk
SEQ =	Sequence number used as the eight digit Litho code. To identify SIDPERS records a letter (B - Bragg, C - Campbell, J - Jackson, P - Polk) was placed in the units position of the number.

5.2.3 SIDPERS OUTPUT RECORD

The output record from the CONVERT program is

<u>COLUMN</u>	<u>FIELD</u>	<u>DESCRIPTION</u>
1-3	PROGID	Program ID (200)
4-12	SSN	Social Security Number
13-14	FMP	Family Member Prefix
15-20	DOB	Date of Birth (yymmdd)
21-23	CAT	Category (A10)
24-31	SEQN	Litho followed by letter
32-37	DAT	Registration date (yymmdd)
38	SEX	Sex
39	RACE	Race
40	MPC	Military pay code prefix
41	blank	
42-43	GRADE	Pay grade
44-46	JC	Job code
47	W	Last character of Job code
48-53	UPC	Location
54-56	blank	
57-61	ZIP	Zip code
62-73	blank	

This output record contains blank fields which make the record format consistent with the output records created by scanning the patient registration forms. Thus the same program can be used when loading patient data into the FOCUS data base regardless of the raw data source, that is, from ACDB forms, SIDPERS or DEERS.

5.3 DEERS

The DEERS data was sent in an attempt to capture retired personnel and dependents as well as to update movement in active duty personnel. DEERS tapes were created by selecting registered DEERS individuals within a fifty mile radius of each site. The fifty mile radius was determined by the zip code in the address of the individual. The DEERS data was sent quarterly from Department of Defense, Manpower Data Center, Monterey, CA 93940.

The large number of records on the DEERS tapes, particularly for BAMC, caused the patient registration file to exceed available capacity. As mentioned before, various ways were discussed to resolve the patient registration problem, but no solutions were found. The DEERS tapes contained many people that never made a hospital visit, so while the tapes increased the percentage of patients registered, they created a great deal of unused information to be stored.

Two possible remedies are to compare Social Security numbers from DEERS data with actual patients and accept only matches or to use a smaller geographical area for selection of DEERS records. In the first instance, once every quarter the data would be brought up-to-date but during the quarter registration data would lag a bit. The primary disadvantage of this method is the amount of computing needed; it was for this reason it was never tried. The sites personnel had too much to do during most of the study to attempt this procedure. This method would have required that the tapes be sent to Ft. Detrick to be read, downloaded to Bernoulli cartridges, and then returned to the sites. The sequential DEERS data on the cartridges would have had to be loaded into the patient Focus file. Programs would have had to be written to perform the matching and selecting these matches for inclusion in the patient file.

An analysis could be made as to how much patient registration increased based on the number of miles from the site. A 50-mile radius was arbitrarily selected but perhaps a 40-mile radius would be adequate. A tape of zip codes with mileage from the sites was sent to Ft. Detrick for analysis near the end of the data collection portion of the study. As a result no analysis of zip codes was done.

5.3.1 DEERS TAPE FORMAT

The format of the DEERS tape is given below. The DEERS data contained more fields than were needed for the ACDB Study. Therefore, only those fields used are provided. While the DEERS tapes were essentially a copy of existing DEERS tapes, an additional field, the UIC for active Army personnel, was inserted over the top of an address field.

<u>POSITION</u>	<u>SIZE</u>	<u>DESCRIPTION</u>
1-9	9	Social Security Number
11-12	2	Dependent Sequence Number
40-45	6	UIC for Army Active Duty
107-111	5	Zip code
116	1	Branch
117-118	2	Paygrade
119	1	Sex
121-126	7	Date of Birth
127	1	Sponsor Status

5.3.2 DEERS FIELD DEFINITIONS

<u>DEPENDENT SEQUENCE</u>
01-19 Child

20	Sponsor
30-39	Spouse
40-44	Mother
45-49	Father
50-54	Mother-in-law
55-59	Father-in-law
60-69	Other

SPONSOR STATUS

A	Active Duty
B	Recalled Active Duty
C	Civilian
D	DAV
E	MEPCOM
J	Academy Student
R	Retired
S	Serv. Sec. Designee
V	Reservist on Active Duty
Z	Unknown

BRANCH

A	Army
E	Public Health
F	Air Force
I	NOAA
M	Marine
N	Navy
P	Coast Guard
X	Other
Z	Unknown

5.3.3 DEERS CONVERSION

As with the SIDPERS tapes, a number of fields require conversion to conform to the fields defined by the Patient Registration Forms. A SAS program called DEERS in the partitioned data set DXB.ACDB.CNTL is used to convert the DEERS tapes.

The conversions performed are as follows:

PAYGRADE

<u>GRADE</u>	<u>CONVERT TO</u>
01-09	E01-E09
10-14	W01-W04
21-30	O01-O10
44-49	E04-E09

DATE OF BIRTH

<u>DOB</u>	<u>CONVERT TO</u>
yyymmdd	yyymmdd

UIC CODES

The sponsor status was used to determine whether or not an individual was active duty. If the individual was active duty then the UIC was taken from the record. Otherwise, blanks were used for the UIC. The following are sponsor codes for active duty:

<u>CODE</u>	<u>DEFINITION</u>
A	Active Duty
B	Recalled Active Duty
E	MEPCOM
J	Academy Student
N	National Guard Active Duty
V	Reservist on Active Duty
Z	Unknown

CATEGORY

Category was determined by Sponsor Status, Branch and dependent sequence. Conversion is shown in the following table:

FOR DEPENDENT STATUS = 20

<u>BRANCH</u>	<u>SPONSOR STATUS</u>					
	A	B	J	N	R	V
A	A10	A10	A70	A26	A30	A23
E	P10	P10	X70	X70	P30	P20
F	F10	F10	F70	F20	F30	F20
I	O10	O10	X70	X70	O30	O20
N	N10	N10	N70	X70	N30	N20
M	M10	M10	X70	X70	M30	M20
P	C10	C10	F70	X70	C30	C20

FOR DEPENDENT STATUS NOT 20

<u>BRANCH</u>	<u>SPONSOR STATUS</u>					
	A	B	J	N	R	V
A	A50	A50	X70	A50	A60	A50
E	P50	P50	X70	X70	P60	P20
F	F50	F50	X70	X70	F60	F20
I	O50	O50	X70	X70	O60	X70
N	N50	N50	X70	X70	N60	N20
M	M50	M50	X70	X70	M60	M20
P	C50	C50	X70	X70	C60	C20

ALL OTHER CATEGORIES ARE X70 EXCEPT:

If Sponsor Status is C then Category is X30
If Sponsor Status is D then Category is K10

If Sponsor Status is S then Category is X20
If Sponsor Status is E and Dependent Status
is 20 then Category is X20

OTHER ADDED FIELDS

SEQ = Sequence number used as the eight digit
Litho code.
RACE = Blank
DATE = Date tape was converted

5.3.4 DEERS PROCESSING

After DEERS data was converted to a compatible format with the ACDB it was sorted by Social Security number and FMP. The next step depended on whether the data was to be added to the mainframe data base or sent to the sites for inclusion in their data base. A program called VSORT in the partitioned data set (PDS) DXB.ACDB.CNTL was used to sort.

Mainframe processing at this step continued with the program PATD32 in the PDS DXB.ACDB.CNTL. This program encrypted the Social Security number and added the site identification code to each record. The data was then loaded into the FOCUS patient file by the FOCUS procedure PATILOAD or PATILD located in the PDS DXB.FOCEXEC.DATA. The PATILOAD program was the FOCUS procedure originally designed to load the Patient Registration forms. It did not update existing records. PATILD procedure was a modified version of PATILOAD which allowed for information to be updated within a patient's record. The DEERS data was constantly updated because of moves or changes in status. The continual updating created another dilemma. Should the record contain current information regarding patient demographics, or should it contain information which was correct at the time of a particular visit? To have correct information at the time of each visit would require multiple records for many patients. This would make the patient file even larger and would require a date key for access.

Collecting the patient's information at every visit and including it with the visit data would have been one way to overcome this problem. This would have increased considerably the time and effort required to collect this information, thus defeating one of the objectives of the study - making the data collection as painless as possible. Since patient registration was a problem during the study, this solution was not feasible.

Processing for the PCs continued by downloading the converted and sorted DEERS data onto Bernoulli disk cartridges which were sent to the sites for loading into each site's FOCUS patient files.

Downloading procedures are in the Mainframe section. Also note that all programs or FOCUS procedures are listed in the appendices.

6.0 FORT DETRICK PROCESSING

Development began at HCSCIA at Ft. Sam Houston in San Antonio, Texas. About June 1985 the programmer left the project and a replacement was obtained. The development work was more than one programmer could handle because of the pressure to get the sites up and running. In August 1985 a second programmer from Ft. Detrick, already tasked under the Performance Measurement Study, was asked to help. By September some sites had had equipment installed and programs were ready to scan a few forms. The second programmer then left the project on 30 September 1985. The Ft. Detrick programmer continued development, working at HCSCIA about 50 percent of the time for the next six months. Eventually a PC system with the scanner was installed at Ft. Detrick and development work remained there. A second PC system was purchased at Ft. Detrick for uploading data.

Presently, all ACDB accumulated data is stored on a mainframe computer at Ft. Detrick in Frederick, Maryland.

6.1 PC PROCESSING AT DETRICK

Two PC systems were eventually installed at Ft. Detrick. One system with an Iomega Bernoulli disk cartridge system and a Model 7001 NCS scanner was used for development. The second system was used for uploading data sent in by the sites to the mainframe computer.

6.1.1 UPLOADING DATA

Several approaches to uploading data from a PC to the mainframe computer were explored. The goal was to be able to upload data in the fastest time possible. Details of each approach are not noted here. Technology is constantly being developed that makes old techniques obsolete. The final uploading approach used an IRMA board manufactured by Digital Communications Associates, Inc. Uploading was achieved at a rate of 2 million bytes per hour. Today, there are methods that are more than twice as fast.

The monthly visit data sent by the sites ranged from 3 to 6 million bytes, thus uploading took from 1.5 to 3 hours per site. With SIDPERS and DEERS data being uploaded and downloaded, the second PC system was used about 40 hours a month for data transfer.

6.1.1.1 IRMA

When installed in a PC, an IRMA board allows bisynchronous communication between the PC and a mainframe. Software included with the IRMA board enabled the PC to simulate a mainframe terminal. FTTSO, a file transfer program, enabled data transfer between the TSO operating system on the mainframe and a PC. Also needed on the mainframe was a resident program provided by Digital Communications Associates at no charge.

FTTSO, a menu driven program, permitted a PC file name and a mainframe file name to be entered. Data Control Block (DCB) information could also be entered letting TSO automatically allocate the mainframe file. Menus containing DCB parameters for visit data, patient data, provider data, clinic data, SIDPERS data and DEERS data were stored so that production control personnel could follow established procedures for uploading data.

Details of the uploading and downloading procedures need not be included in this manual since IRMA documentation is available; if the study were to be started up again, different techniques would be used. An example of the FTTSO menu is provided below with the Transfer Parameter section included.

IRMALink FT/TSO Ver. 2.10

Copyright 1984 Digital Communications Associates

PC filename:

Data set name:

Transfer type:

Function Keys		Transfer Parameters	
1 SEND	2 HELP	Block size:	Transfer mode:
3 RECV	4 ERS INP	Record size:	Expands TABs:
5SAVE TY	6 ERS EOF	Record format:	Truncate spaces:
7 RESTAR	8 NXT VAL		U/L case:
9 EXIT	10 RESET	Line numbers -	Volume serial:
		Position:	Space Units:
		Length:	Primary qty:
		Code table:	Secondary qty:
			Directory blocks:

The PC file name is entered in the first line and the mainframe file name in the second line. The last part of the mainframe file name automatically is repeated on the third line for transfer type if a menu for the transfer parameters has previously been created and stored. Thus the transfer parameters are invoked by the mainframe file name. For example, visit data from BAMC had a mainframe file name of PMS.BAMC.JAN87.VISIT. The last part of the name, VISIT invoked the visit transfer parameters automatically when the name was entered. The operator then used the F1 function key to send data to the mainframe. For downloading, the transfer parameters were unnecessary.

6.1.2 DEVELOPMENT

Development of PC systems for the ACDB Study originally began at HCSCIA at Ft. Sam Houston, Texas. A IBM XT with 256K memory, a 10M byte hard drive and an Iomega Bernoulli 10+10 external drive was delivered to Ft. Detrick, Maryland, early in the study. The Ft. Detrick system was to be used generally for uploading data. Specific details of how this was to be done had to be formulated. After the first two programmers at HCSCIA left the study, a Ft. Detrick programmer took over the software area of the project. For several months, September 1985 to March 1986, half of the programming effort was done at HCSCIA using the IBM XT and NCS Scanner Model Sentry 7001. FOCUS, the DBMS was also available on this system. Eventually, the Ft. Detrick system was expanded by adding FOCUS, which included a memory board and brought the total memory to 768K bytes. The NCS scanner was shipped from HCSCIA to Ft. Detrick, providing a total system capable of all the software development needs of the study.

Early in the study the primary goal of system development was to have a minimum system capable of scanning forms and storing data. Most of the Basic programs to scan forms were written but not tested. During the change of programmers, the fact that programs were untested was overlooked until the sites began to have problems in scanning. Correctly completed forms were rejected. Additionally, some data was not collected properly, and these errors made their way into the data base. It wasn't until December of 1985 that most of the bugs were worked out. All data collected prior to January 1, 1986, has been set aside and will not be used in analysis.

During this time there were still new programs being written as new forms were being designed. FOCUS

programs were also needed to give the sites reporting capabilities. To aid in the success of the project, an attempt was made to provide a useful product early in the study. Two FOCUS contractors from IBI were hired primarily to develop a series of FOCUS reporting programs. Twelve general reports were considered to be appropriate, and about eight were completed. In addition, FOCUS courses were given by the contractors to ACDB personnel.

Development continued through 1986. The programmer also provided technical support to local study sites during this time. A few new forms were designed, and programs had to be written for them. No fewer than ten updates were sent to the sites during 1986. Some included new description files, some new programs and some fixes to problems. There was also a problem of time. It was soon discovered that FOCUS took a good deal of time to load data and produce reports from the monthly data. Tuning the system to make it more efficient became a critical requirement. Site operators were not able to keep up with scanning forms as the numbers received from the clinics increased. There were not enough hours in the day to scan, load data and run reports. Several remedies were taken.

One remedy was to relax some of the rigorous edits on each of the forms. Although some data elements might not be as clean, the forms would scan faster, and less time would be needed by the site operators to sort through the rejects and correct them or send them back to the clinics for correction. Relaxed edits probably resulted in more data collected even though some of it may be in error. Under rigorous edits some forms returned to the clinic for correction never made their way back to be input; thus the data for the entire form was lost.

To relax the edits, resolution sheets for each form had to be changed and tested. To save time the first five pages of the scanner resolution sheets were shared among all the forms. These five pages were nearly identical except for certain forms. Therefore an exception had to be made on these pages and care taken to insure the pages were coded for the proper form. Unavoidable errors caused problems at the sites; these problems had to be resolved. Solutions had to be tested and another update sent out. By the end of 1986 all pages of the resolution sheets were completed, and these kinds of errors were eliminated.

Development efforts to improve efficiency continued

throughout the year. Most FOCUS report programs were rewritten entirely to make them more efficient. A night reporting system (beginning in late 1985) allowed report programs to be queued and run at night. Since the printer needed to be monitored in case of jams, the night reports were printed out during the day. Experiments changing the location of FOCUS and data bases on the disk drives were made. Most changes of this nature caused more operational problems negating any benefit of time saved.

Another development effort during 1986 was the implementation of the SIDPERS and DEERS data bases for patient registration.

Throughout the year FOCUS was being improved, and two upgrades were received and installed at the sites. If a system is working smoothly, it is not necessary to install upgrades, but there were major problems in each FOCUS version or upgrade that made it desirable to install improved versions. Although some of the old problems were solved, new problems occurred. These new problems took some effort to trace the causes and correct them. As time went on, there were fewer FOCUS problems (or techniques were found to circumvent them). One problem which occurred late in the year caused the computer to freeze. The only solution was to turn the computer off and then back on. Of course, the operator had to reenter FOCUS and start over again with whatever task was being performed. The cause was partially the poor memory management portion of FOCUS, but the problem was overcome by upgrading to the next level of DOS. Although the problem did not entirely disappear, it was manageable.

Disk space became critical as patient registration files grew rapidly with the SIDPERS and DEERS data being used to supplement form registrations. This and the desire for more efficiency brought about the decision to upgrade the hardware at each site. An additional PC system was installed at each site except Redstone Arsenal. One system was devoted to scanning, and the other was used for data base operations such as loading, reporting and preparing data for shipment. Development effort was required to split the single system into two systems. Along with this hardware installation, the upgrading of the Imomega Bernoulli 10M byte drives to 20M drives was done. The initial site to upgrade to new hardware was Ft. Campbell in September 1986.

The last major effort in development occurred in the first five months of 1987. All forms except for the

registration forms were completely redesigned. This meant all resolution programs had to be rewritten along with all the Basic programs used to scan each form. FOCUS changes were required since some fields in the data base were no longer collected, and there were some new fields. A temporary employee was hired at Ft. Detrick during this time to help in the redesign development. New techniques were used particularly in the scanning portion. Rather than code the resolution sheets by hand, a program was obtained from NCS to code them by data entry on the computer. Continuous resolution sheet forms were obtained, and the new sheets were printed directly from the computer using special ribbons. The advantage of this method was that the great amount of duplication of fields on many forms could be handled easily by computer, saving many hours of hand coding. The disadvantage was that the printing of the resolution sheets was critical, and in many cases extra time was spent getting the scanner to read them. New techniques available now allow the entire process to remain in the computer, eliminating the need for resolution sheets and scanner programs.

The natural course of events after any major change (or minor for that matter) is to work out the bugs after installation. Experience and more thorough testing kept problems to a minimum the second time around.

Development effort essentially ended by July of 1987 as far as the data collection portion of the study was concerned.

6.1.3 TECHNICAL SUPPORT

Technical support was also provided by Ft. Detrick. Technical support included aid in diagnosing hardware and software problems on the scanner and IBM XT systems. Most problems were solved or their causes determined by telephone. On a few occasions listings or diskettes were sent to Ft. Detrick for analysis. One or two cases required a site visit.

Support by telephone has its disadvantages. Solving programming and hardware problems depends on visual clues. Two approaches can be taken, and the final solution is usually a combination of these approaches. The first approach is to examine the symptoms and obtain clues that point to another area which may be causing the problem. The second is to eliminate areas of possible causes by examining symptoms and/or testing them independently. Either method depends on a logical step-by-step process. Directories,

listings, and error messages are types of data that must be scanned. Usually the eye of an experienced programmer can pick out an unusual sequence of characters or some other clue from listings or screens of data. This cannot be done over the telephone. Data must be read. Instructions must be given to execute DOS or other commands that may not be familiar to the operator. Commands usually are required to be spelled since they are not English words. Then letters such as 's' or 'f' are easily misunderstood. Debugging by telephone can be a long and tedious process.

Testing on another machine by running the same task or examining copies of programs is also helpful. A telephone headset was purchased to allow the hands to be free to test options while continuing to talk on the telephone. Time spent on the telephone each week probably averaged 15 to 20 hours. In many cases programming problems, if found at one site, were resident at all sites. Thus each site had to be contacted and warned. Usually this meant not scanning a particular form or not running a particular report, and it did not stop the entire operation. In some cases the solution could be given over the telephone but in many cases update diskettes had to be sent out. Update diskettes were usually sent out on a monthly basis.

It was through the cooperation and patience of site personnel that so much was accomplished by telephone. At the beginning of the study many operators were weak in DOS and FOCUS, and their willingness to listen to step-by-step procedures and explain results over the telephone was admirable. As the study progressed, the operators' experience and knowledge of the equipment and system programs increased, making debugging easier and less time consuming.

Describing the details of all the problems is not productive, but some items may be of interest. A maintenance contract with NCS provided servicing; if any hardware failure occurred, it was to be repaired or replaced. The field engineers, while experienced in scanner maintenance, were weak in the IBM and Iomega equipment. Many times the field engineer had to be told what to replace after our own diagnosis. When the additional IBM systems were added in the fall of 1987, many of the field engineers called Ft. Detrick rather than NCS to ask about the installation procedure. There were some sticky scanner problems, and in some cases the wrong part was sent. Overall, NCS provided good support, and equipment was usually replaced or repaired within 24 hours.

Feeding of the sheets into the scanner caused some early problems until a scanner procedure called 'double sheet calibration' was discovered. This procedure calibrates the double sheet indicator and is unique for each scanner. For some reason the site operators did not know about this procedure and put up with considerable grief trying to scan without jams. It was either through some operator reading the manuals or a field engineer making a suggestion that this procedure solved most of the jamming problems.

Technical support was provided throughout the study along with development. While time consuming, it was aided by the cooperation of the site personnel.

6.1.3.1 OPTIMIS

OPTIMIS (Operation Management Information Management System) of the Department of the Army provided an electronic mail capability. Each site had a built-in modem which could be used to dial into OPTIMIS.

The electronic mail facility was very helpful in maintaining communications among the sites, HCSCIA and Ft. Detrick. It was not always possible to reach personnel by telephone because of time zones and varying work hours. It was desirable to reach the sites as quickly as possible when a system error affected all sites. OPTIMIS provided a distribution list capability which allowed messages to be sent to all sites simultaneously.

The advantages of OPTIMIS were (1) messages could be sent at any time (2) messages could be read when convenient (3) messages could be sent simultaneously to all sites by distribution lists (4) written instructions could be sent making interpretation clearer (5) written instructions could be studied. and (6) the messages could be printed and saved for later reference. While the advantages of providing technical support by telephone have been expounded above, electronic mail also was critical and helped considerably in resolving and correcting problems.

An OPTIMIS account was obtained for all site personnel and for most of the HCSCIA staff working on the project. Communications between HCSCIA and Ft. Detrick were made more convenient by OPTIMIS.

6.2 MAINFRAME COMPUTER

The mainframe computer at Ft. Detrick is an AMDAHL

470/V8 (IBM 3033 compatible) with 16 megabytes of memory and 16 data channels. It is capable of executing 6.9 million instructions per second.

Direct access storage is constantly growing. At the time of the data collection portion of the ACDB Study there were six IBM 3380 units with 2.5 gigabytes each. In addition there were 16 IBM 3350 disk drives with 317.5 megabytes each as well as 16 IBM 3330-11 units with 200 megabytes each.

There are eight STC 4670 tape units with 1600/6250 BPI capability as well as two IBM 2401 units with 800/1600 BPI capability. Two IBM 1403 printers are on line, capable of printing 1100 lines per minute each. A COMTEN 3690 communications controller with 32 bisynchronous ports and 96 asynchronous ports is also available.

6.2.1 TSO/ISPF ENVIRONMENT

Several operating systems reside on the mainframe computer. The Time Sharing Option (TSO) is used by the study. It is an interactive terminal system which allows the user conversational access to the facilities of the computer. Knowledge of TSO commands is necessary to operate within this system.

Under TSO is a package called ISPF (Integrated System Productivity Facility) which extends the capabilities of TSO. ISPF has full screen editing commands with split screen capability. Most of the TSO commands are presented as menu options. The menu options make the system more user friendly and make learning the TSO commands unnecessary. Basic familiarity with the operating system and file structures is necessary if utility and development efforts such as setting up FOCUS areas are to be carried out.

Online sessions as well as batch jobs may be run. FOCUS may be used in both modes. JCL (Job Control Language) knowledge is necessary for submitting batch jobs.

There are several program languages and data base management systems available on the mainframe computer. Two such systems used by the study are SAS and FOCUS.

It is not the purpose of this manual to give a detailed explanation of TSO, ISPF, or JCL. There are IBM manuals available on these subjects. There are also

manuals for SAS and FOCUS. However, where detailed explanations may help in understanding how the ACDB data processing procedures are used within the system, they are given.

6.2.2 MAINFRAME FOCUS

Mainframe FOCUS is similar in many ways to PC FOCUS particularly in writing report requests. However, there are many differences that can cause problems for the casual user without mainframe experience. Differences between mainframe FOCUS and PC FOCUS are not explained here, but explanations of how to apply FOCUS techniques for applications used in the ACDB Study are furnished.

FOCUS runs under the TSO ISPF operating system environment. Thus a knowledge of some TSO is necessary for the FOCUS user. TSO commands may be used within FOCUS procedures. Other maintenance activities require TSO intervention. Debugging and monitoring programs can be done through TSO. Batch jobs require JCL knowledge. In a thoroughly developed system where all programs are written and tested, the user can be sheltered from much of the background knowledge such as TSO or JCL. In this study where the design characteristics evolved throughout the data collection period and where analysis of the data led in different directions depending on the outcome of the results, a closed system was not developed. Thus the researcher using the data must be familiar with mainframe operating systems as well as FOCUS. In addition he or she must be aware of the data collection techniques and problems of the study in order to analyze the data and assess the results properly.

There are three file types in FOCUS, and on the mainframe these files are stored in different areas. There must be a file description, the data file, and the FOCUS procedures or programs. The PC distinguishes the type of file by assigning an extension for each type. The mainframe stores master files and procedures in partitioned data sets (PDS). The PDS for master files is called PMS.MASTER.DATA. The PDS for FOCUS procedure files is called FOCEXEC.DATA. The FOCEXEC.DATA PDS is user specific. In other words each user has his own FOCEXEC PDS. The PDS name is preceded with the user's ID, for example: DXB.FOCEXEC.DATA. The master and data files are common to all users and have a general user prefix. The general user prefix, PMS, stands for Performance Measurement Study, of which the ACDB Study is a part.

A PDS is similar in some ways to a subdirectory on

a PC. Many files may reside within a PDS and are called members. The data base files are not in a PDS and the structure parameters must correspond to the requirements of FOCUS. The names of the data base files end in FOCUS. An example of the visit file for BAMC is PMS.VBAMC.FOCUS.

6.2.3 FOCUS PROGRAMS

Most of the FOCUS procedures written for the mainframe are data loading programs. These are FOCUS modify procedures and are very similar to the PC versions in many cases. Reporting procedures written for the mainframe are mostly ad hoc programs written when some analysis of the data is required. The FOCUS procedures are members of a FOCEXEC.DATA partitioned data set.

6.2.3.1 LOADING PROGRAMS

FOCUS Modify procedures were used to load raw data into FOCUS structured hierarchical files. The mainframe load program loaded visit data into FOCUS files sent in each month from the sites similar to the PC programs that loaded scanned data into FOCUS files. Other data which needed loading was the patient registration data. This was sent periodically from the sites and supplemented by the SIDPERS and DEERS data. Provider registration data from the sites was also periodically updated. Code descriptions such as Clinic descriptions, Diagnosis descriptions, Procedure descriptions and Other code descriptions initially required loading and occasionally updating to incorporate corrections and additions.

Listings of the loading programs are provided in Appendix E. A visit loading procedure was written for each site. They are the same except for the input and output file names. The appendix includes an example of a visit procedure for the new data (after the form revision in May 1987). The data at each site was dumped by a FOCUS Table Request which created a file nearly identical to the raw data initially supplied. Thus the visit load procedure for the mainframe was similar to the PC version except for the areas concerning computer architecture.

Listings provided in the appendix are

LBAMC	Loading procedure for visit data
PATILoad	Loading procedure for patient data
PROVLOAD	Loading procedure for provider data

PROCLOAD Loading procedure for procedure code
descriptions
DIAGLOAD Loading procedure for diagnosis code
descriptions
OTHLOAD Loading procedure for other code
descriptions

6.2.3.2 REPORTING PROCEDURES

Reporting procedures are called Table Requests by FOCUS. No mainframe Table Requests are included since no standard procedures were generated for the study. A warning may be given here as may have been stated elsewhere in this manual.

FOCUS, a fourth generation language, makes it easy to write Table Requests. However, without a knowledge of the hierarchical structure of the data base and the behind the scenes operation of FOCUS statements, an incorrect report can be created more easily than a correct one. An experienced programmer tests procedures on known data to insure accuracy before releasing results from a data base. The simplicity of FOCUS Table Requests lures the novice into a false sense of accomplishment when sophisticated-looking reports are generated even though the results may be inaccurate.

Another caveat concerns execution time. On a large data base, a FOCUS procedure may take hours to run. The large amounts of time required for loading and running reports on the PC have already been mentioned. Even though the mainframe is much faster than the PC, procedures still may take hours to run. One reason for this is that the data base is now much larger, consisting of data from the six sites accumulated over the life of the study. Another cause is inefficient writing of procedures. As in all programming languages, there is more than one way to do a given task. When there are multiple ways to do a task, one way is usually more efficient. Knowledge of FOCUS internals helps in designing efficient procedures.

6.2.3.3 RUNNING BATCH PROGRAMS

Running a FOCUS procedure on a large data base can be time consuming. It is not convenient to attempt to run large jobs on-line. Further, some procedures generate large reports which sometimes can be printed more conveniently through a batch job. A batch job is run in a background time-sharing mode. When a FOCUS job is submitted, various work areas must be reserved (allocated). In addition, an assessment must be made of

the data bases needed for a report. The JCL must be written to convey to the computer all the information necessary to carry out the reporting task. An example of a batch job JCL to run a FOCUS procedure is given below:

```
//DXB1Z JOB P101,BOLLING,CLASS=F,MSGCLASS=X,MSGLEVEL=(2,0)
//STEP1 EXEC PGM=FOCUS,REGION=4096K
//STEPLIB DD DSN=SYS2.FOC503.FOCLIB.LOAD,DISP=SHR
//ERRORS DD DSN=SYS2.FOC503.ERRORS.DATA,DISP=SHR
//FOCSTACK DD UNIT=SYSDA,SPACE=(TRK,(1,1))
//FOCSORT DD UNIT=SYSDA,SPACE=(CYL,(6,6))
//HOLDMAST DD UNIT=SYSDA,SPACE=(TRK,(1,1,1))
//HOLD DD UNIT=SYSDA,SPACE=(CYL,(20,20))
//SYSPRINT DD SYSOUT=*
//OFFLINE DD SYSOUT=*
//MASTER DD DSN=PMS.MASTER.DATA,DISP=SHR
//FOCEXEC DD DSN=DXB.FOCEXEC.DATA,DISP=SHR
//VBAMC DD DSN=PMS.VBAMC.FOCUS,DISP=SHR
//VBRAG DD DSN=PMS.VBRAG.FOCUS,DISP=SHR
//VCAMP DD DSN=PMS.VCAMP.FOCUS,DISP=SHR
//VJACK DD DSN=PMS.VJACK.FOCUS,DISP=SHR
//VPOLK DD DSN=PMS.VPOLK.FOCUS,DISP=SHR
//VREDS DD DSN=PMS.VREDS.FOCUS,DISP=SHR
//CLINIC DD DSN=PMS.CLINIC.FOCUS,DISP=SHR
//DIAGNOS DD DSN=PMS.DIAGNOS.FOCUS,DISP=SHR
//PROVIDER DD DSN=PMS.PROVIDER.FOCUS,DISP=SHR
//PROCEDUR DD DSN=PMS.PROCEDUR.FOCUS,DISP=SHR
//SYSIN DD *
SET PASS=XXXXX
SET BINS=64
EX AZT
FIN
/*
//
```

Statements beginning with // or /* are JCL statements. The first statement is the JOB card (carried over from the days when jobs were submitted on punched cards). This identifies the user, gives accounting information, and tells the computer other information needed to run the job.

The next three statements assign a work region and call FOCUS. The following four statements allocate disk space for FOCUS work areas. Some critical space areas are FOCSORT and HOLD. A procedure that produces a large report may require additional disk space for sorting. Other procedures may create temporary files called HOLD files. Should these be large, additional space may need to be allocated. Notice these lines have a SPACE parameter which is used to allocate disk space in tracks

or cylinders.

The other DD statements are mostly statements that allow FOCUS to recognize data bases required for the FOCUS procedure being executed. In the example, visit data for each site (VBAMC, VBRAG, VCAMP, VJACK, VPOLK and VREDS) as well as some of the description files, CLINIC, DIAGNOS, and PROCEDUR are mentioned. If the file is to be used in the procedure, it must be mentioned in the JCL; however, if a file is mentioned and not used, no harm is done. A generic JCL file can be maintained which includes all the input data base files. This reduces editing effort every time a new batch job is needed.

The last few lines with no // or /* at the beginning are FOCUS statements. These are identical to on-line FOCUS statements. In the example the first line sets the password. This is critical should the data base files be protected by a data base administrator (DBA). No data will be read unless a correct password is given. The second SET statement assigns the number of BINS. The maximum number is 64. BINS are work areas or buffers that FOCUS uses for I/O (input/output tasks). Where space is available, a maximum setting usually is most efficient. The next statement, EX AZT, requests that the procedure AZT be executed. Lastly a FIN statement exits FOCUS.

The previous example can be used for most batch jobs but certain procedures may require additional JCL or modifications to statements given. If a FOCUS procedure uses a FOCUS library routine, the following lines are needed:

```
//USERLIB DD DSN=SYS2.FOC503.FOCLIB.LOAD,DISP=SHR
//          DD DSN=SYS2.FOC503.FUSELIB.LOD,DISP=SHR
```

In cases where data is to be appended to a hold file, DISP=MOD should be added to the HOLD statement. Care should be taken to remove this modification in procedures that do not require it.

Batch files are usually prepared in a PDS other than a FOCEXEC.DATA PDS. However, they can reside in the FOCEXEC area. Batch files may be submitted by bringing the file up under the TSO editor or in the FOCUS editor (TED) and issuing the submit command (SUB).

6.2.4 SAS PROGRAMS

SAS, which stands for Statistical Analysis System,

is a product supplied by SAS Institute Inc. In addition to its strong statistical capabilities, it can be used for many utility and maintenance operations that supplement the TSO operating system. A detailed explanation of SAS is not given in this manual as there are a number of SAS manuals available.

SAS is very powerful in its statistical capability. It can therefore complement FOCUS when sophisticated statistical analysis is required. FOCUS can create sequential output files with subsets of a data base which can then be used as input files to SAS.

SAS has been used to inspect and repair damaged files received from the sites.

6.2.5 DEERS AND SIDPERS PROCESSING

Two SAS programs have been written that have been used on a regular basis. These programs convert the SIDPERS and DEERS data and format it for input into the FOCUS Patient data base. A description of SIDPERS and DEERS data usage is presented under the Patient Registration section in Special Topics of this manual. The two programs are SIDCONV and DEERS. Listings of the programs are in the appendix.

SIDPERS tapes were received and copied to disk under the name DXB.site.mmmmy.SIDATA. Site is a four letter code representing the ACDB site from which the data was received (BAMC, BRAG, CAMP, JACK, REDS, or POLK). The three character month abbreviation and last two digits of the year were inserted in the mmmmy field. The data was sorted by Social Security number. A copy of the data was downloaded for PC processing. The mainframe processing continued by converting the data using the SIDCONV program. SIDCONV converted the data and added a site identification. It also encrypted the Social Security number. The output was compatible with the FOCUS patient load program, PASILO. PASILO loaded the data into the patient file, PATIENT.FOCUS. A listing of PASILO is in the appendix.

DEERS tapes were received and copied to disk. The data was converted by the DEERS SAS program and sorted by Social Security number. The conversion program converts DEERS data fields into fields compatible with the ACDB study. A site identification code, as in the SIDPERS files, is added. At this stage the data is downloaded and sent to the sites for loading into the PC data bases. Mainframe processing continued with Social Security number encryption and loading into the FOCUS

patient file using PATILOAD. A listing of DEERS is in the appendix.

6.2.6 DOWNLOADING

Downloading of files from the mainframe to a PC was accomplished using the IRMA FTTSO program. A mainframe program provided the linkage needed to access mainframe files. A description of FTTSO is given in the PC Uploading section of this manual. Downloading was simpler than uploading as no data control block parameters were needed. All that needed to be done was to invoke the FTTSO program, enter the PC and mainframe file names, and hit the F3 function key to receive the data.

6.3 SITE ACCESS TO MAINFRAME

Limitations of disk storage space on the PCs at each site prevented easy access to more than one month's data. Several sites could see an advantage for doing longitudinal reports. Some providers liked a record of the number of patients seen and the reasons for seeing them. There were two ways this could be accomplished using the accumulated data on the mainframe.

Requests for reports could be made by telephone or electronic mail. This was done a number of times for several sites. The report program was run against the accumulated data for a particular site and the results mailed back to the site. If the report was a summary type report and consisted of a few lines, the results could have been sent back by electronic mail. Because there was only one person at Ft. Detrick to provide technical support to all the sites in addition to his regular system development duties, the amount of time available was limited. Therefore, mainframe access by each site was deemed necessary.

Each site already had dial-up capability with Crosstalk, a modem, and an Optimis account for electronic mail communications. The mainframe computer had to be set up for site access. This required a FOCUS Exec file for each site. The FOCUS Exec file was a storage area that contained the FOCUS report procedures. FOCUS automatically accesses a user's Exec file for the programs it needs. The master file descriptions were already available and were shared by each site. Other than accessing the data which was already available, no other FOCUS consideration was necessary.

Each site was assigned a Data Base Administrator (usually the site operator) and a password to the mainframe computer. Because the site personnel had no training or experience in using mainframe operating systems, a CLIST (command list) was created to provide direct access to the FOCUS system. The CLIST was a group of TSO commands that allowed various tasks to be executed under the mainframe operating system. This CLIST was set up so that when the mainframe computer was accessed from a site, all the FOCUS files necessary for that site were allocated, and the operating system placed the user directly into the FOCUS environment. Here the user was familiar with many of the FOCUS commands and editing techniques because of the similarity to the PC version of FOCUS. Only minimal assistance was needed to overcome some of the differences between the PC and mainframe. When the user exited from FOCUS, the CLIST regained control and logged the user off the mainframe. Thus the operating system was transparent to the user. An example of the CLIST for Ft. Polk follows:

```

PROC 0 DEBUG
IF $&DEBUG = $DEBUG THEN +
  CONTROL CONLIST LIST SYMLIST MSG
  ELSE CONTROL NOMSG
FREE F(MASTER FOCEXEC ERRORS USERLIB HOLD HOLDMAST SAVE FOCSORT)
FREE F(REBUILD FOCSTACK FOCXML OFFLINE DIAGNOS PROCEDUR PROVIDER)
FREE F(PATIENT CLINIC VPOLK NPOLK NDIAG NPROC OTHER)
ALLOC F(HOLD) SP(300 60) TRACKS
ALLOC F(HOLDMAST) SP(10 10) TRACKS DIR(10)
ALLOC F(SAVE) SP(300 60) TRACKS
ALLOC F(REBUILD) SP(500 100) TRACKS UNIT(SYSDA)
ALLOC F(FOCSORT) SP(30 10) TRACKS
ALLOC F(FOCSTACK) SP(20 20) TRACKS
ALLOC F(FOCXML) SP(100 50) TRACKS
ALLOC F(OFFLINE) SYSOUT(A)
ALLOC F(MASTER) DA('PMS.MASTER.DATA') SHR
ALLOC F(FOCEXEC) DA('&SYSPREF..FOCEXEC.DATA' 'PMS.FOCEXEC.DATA') SHR
ALLOC F(DIAGNOS) DA('PMS.DIAGNOS.FOCUS') SHR
ALLOC F(PROCEDUR) DA('PMS.PROCEDUR.FOCUS') SHR
ALLOC F(PROVIDER) DA('PMS.PROVIDER.FOCUS') SHR
ALLOC F(PATIENT) DA('PMS.PATIENT.FOCUS') SHR
ALLOC F(CLINIC) DA('PMS.CLINIC.FOCUS') SHR
ALLOC F(VPOLK) DA('PMS.VPOLK.FOCUS') SHR
ALLOC F(NPOLK) DA('PMS.NPOLK.FOCUS') SHR
ALLOC F(NDIAG) DA('PMS.NDIAG.FOCUS') SHR
ALLOC F(NPROC) DA('PMS.NPROC.FOCUS') SHR
ALLOC F(OTHER) DA('PMS.OTHER.FOCUS') SHR
ALLOC F(ERRORS) DA('SYS2.FOC503.ERRORS.DATA') SHR
ALLOC F(USERLIB) DA('SYS2.FOC503.FOCLIB.LOAD' +
  'SYS2.FOC503.FUSELIB.LOAD') SHR
CALL 'SYS2.FOC503.FOCLIB.LOAD(FOCUS)'
FREE F(MASTER FOCEXEC ERRORS USERLIB HOLD HOLDMAST SAVE FOCSORT)
FREE F(REBUILD FOCSTACK FOCXML OFFLINE DIAGNOS PROCEDUR PROVIDER)
FREE F(PATIENT CLINIC VPOLK NPOLK NDIAG NPROC OTHER)
LOGOFF

```

The limitation of this application was the printing capability of the PCs. It was not practical to transmit large reports over telephone lines because of the cost involved and the possibility of errors during transmission. Also, the printers at the sites were not capable of the speeds needed to efficiently handle this traffic. More expensive modems could have handled error checking, and enhanced printers could have printed faster allowing the transmission of reports back to the sites. With existing equipment, however, report requests could be written by the site user and executed with the results then printed at Ft. Detrick and mailed back to the site.

All six sites were given the capability to access

the mainframe data although not all of them used it. There were several reasons for this. In some cases new site operators had all they could handle learning the PC systems. In some cases, the sites had little foresight in regarding the advantages of preparing reports from the data. In all cases, the sites were extremely busy scanning and running their own reports so that little time could be spent picking up an extra task such as accessing and developing additional FOCUS procedures on the mainframe. Had the study continued to the point where the development process halted, and had the sites had adequate personnel to run their systems, more mainframe use would probably have occurred.

6.4 MAINFRAME DATA BASE DESCRIPTION

The mainframe data was different from PC data in several ways. The first difference was size. PC data was collected on a monthly basis. Reports and other programs were run on the monthly data which in the larger sites could be four to six million characters of data for the Visit file. The mainframe data was an accumulation of all the months for each of the six sites and exceeded 186 million characters for the first part of the study. The second difference was in a design change which was implemented in the first five months of 1987 and became effective May 1987. Some old fields were eliminated, and some new fields were created. While the old and new data bases were similar, they were different enough to have to be kept separate. Thus master files and double data files had to be maintained and kept available. The user had to be familiar with the names of these files and use the proper files to access the portion of the study being analyzed. Additionally, since the data from the six sites were combined, a method was needed to identify where the data originated. A new field or extension of some fields was created containing site identification. The last difference had to do with security and confidentiality of the data. Because of the confidentiality of medical data, some fields were eliminated or encrypted when transferred to the mainframe. For instance, provider names were deleted and Social Security numbers of patients were encrypted. FOCUS master file descriptions had to be changed from PC versions to reflect these changes. Additionally, security functions were performed on the data at the site level so that when the data on portable disk cartridges left the site by mail the confidentiality of the data was preserved.

The mainframe data was initially designed to be very similar to the PC data. Thus FOCUS load and report

procedures as well as others could be run on both systems with little modification. There were six data files: a Visit file, a Clinic description file, a Patient Registration file, a Provider Registration file, a Procedure description file and a Diagnosis description file. After several months, the Visit file became unwieldy to maintain. In addition some programs took an excessive amount of time to execute. The Visit file was broken up into six files, one for each site. FOCUS could easily join these files when reports required composite data, and programs ran more efficiently. The names of the files are given below:

PMS.VBAMC.FOCUS	Visit file for BAMC
PMS.VBRAG.FOCUS	Visit file for Bragg
PMS.VCAMP.FOCUS	Visit file for Campbell
PMS.VJACK.FOCUS	Visit file for Jackson
PMS.VPOLK.FOCUS	Visit file for Polk
PMS.VREDS.FOCUS	Visit file for Redstone
PMS.CLINIC.FOCUS	Clinic description file
PMS.PATIENT.FOCUS	Patient registration file
PMS.PROVIDER.FOCUS	Provider registration file
PMS.PROCEDUR.FOCUS	Procedure description file
PMS.DIAGNOS.FOCUS	Diagnosis description file

The PMS preceding each file name is a mainframe project identifier. PMS stands for Performance Measurement Study.

After the redesign of the forms in May 1987, new data files were created in some cases. The data files are generally referred to as old data and new data. When analyzing the old data, the files in the above list should be used. When analyzing the new data, the following files should be used.

PMS.NBAMC.FOCUS	New Visit file for BAMC
PMS.NBRAG.FOCUS	New Visit file for Bragg
PMS.NCAMP.FOCUS	New Visit file for Campbell
PMS.NJACK.FOCUS	New Visit file for Jackson
PMS.NPOLK.FOCUS	New Visit file for Polk
PMS.NREDS.FOCUS	New Visit file for Redstone
PMS.CLINIC.FOCUS	Same as old file
PMS.PATIENT.FOCUS	Same as old file
PMS.PROVIDER.FOCUS	Same as old file
PMS.NPROC.FOCUS	New Procedure descriptions
PMS.NDIAG.FOCUS	New Diagnosis descriptions
PMS.OTHER.FOCUS	Other code descriptions

Notice the new file called OTHER. This file was created because of special codes that were neither procedures nor diagnoses.

6.4.1 MAINFRAME MASTER FILE DESCRIPTIONS

Master file descriptions in the FOCUS format are provided below for the data files listed in the previous section. Since the master file descriptions for the visit file for each site are identical, only one listing is provided.

The master files are stored in a PDS called PMS.MASTER.DATA. Each master file is a member in the PDS. The member names correspond to the portion of the data file names between the periods. Thus the master file corresponding to PMS.VBAMC.FOCUS is written PMS.MASTER.DATA(VBAMC). A member name follows the PDS file name in parentheses.

6.4.1.1 OLD VISIT MASTER FILE

```
FILENAME=VBAMC,SUFFIX=FOC,$

SEGNAME=SITE_ID,SEGTYPE=S1,$
  FIELDNAME=SITE,      ALIAS=SID,      FORMAT=A4,$

SEGNAME=VISLOC,PARENT=SITE_ID,SEGTYPE=S1,$
  FIELDNAME=CLINIC,    ALIAS=CL_UCA,   FORMAT=A4,$

SEGNAME=VISDTE,PARENT=VISLOC,SEGTYPE=S1,$
  FIELDNAME=VISIT_DATE, ALIAS=VDATE,   FORMAT=I6YMD,$

SEGNAME=VISPROV,PARENT=VISDTE,SEGTYPE=S1,$
  FIELDNAME=PROV_ID,    ALIAS=PROV1,    FORMAT=A5,$

SEGNAME=VISITOR,PARENT=VISPROV,SEGTYPE=S1,$
  FIELDNAME=PATIENT_ID, ALIAS=PTID,     FORMAT=A8,$

SEGNAME=LITHO,PARENT=VISITOR,SEGTYPE=S1,$
  FIELDNAME=LITHO_CODE, ALIAS=LITHO,    FORMAT=I8,$
  FIELDNAME=FORM_NUM,   ALIAS=FNUM,     FORMAT=A2,$

SEGNAME=V_DATA,PARENT=LITHO,SEGTYPE=U,$
  FIELDNAME=VISIT_REASON,ALIAS=VISRES,  FORMAT=A1,$
  FIELDNAME=PROV1_TIME,  ALIAS=TIME1,    FORMAT=I3,$
  FIELDNAME=PROVIDER_2,  ALIAS=PROV2,    FORMAT=A5,$
  FIELDNAME=PROV2_TIME,  ALIAS=TIME2,    FORMAT=I3,$
  FIELDNAME=PROV2_RESON,  ALIAS=PROV2RES, FORMAT=A1,$
  FIELDNAME=PROV1_SEEN,  ALIAS=PROV1SEEN,FORMAT=A1,$
  FIELDNAME=PROV1_SAME,  ALIAS=PROV1SAME,FORMAT=A1,$
  FIELDNAME=PROV2_SEEN,  ALIAS=PROV2SEEN,FORMAT=A1,$
  FIELDNAME=PROV2_SAME,  ALIAS=PROV2SAME,FORMAT=A1,$
  FIELDNAME=LAST_12,     ALIAS=LAST12,   FORMAT=A1,$
  FIELDNAME=NEW_PROBLEM, ALIAS=NEWPROB,  FORMAT=A1,$
  FIELDNAME=APPT_STATUS, ALIAS=APPSTAT,  FORMAT=A1,$
```

```

FIELDNAME=INP_REFERAL, ALIAS=REFERAL, FORMAT=A1,$
FIELDNAME=PLACE_OF_VIS, ALIAS=PLACE, FORMAT=A1,$
FIELDNAME=JOB_RELATED, ALIAS=JOBREL, FORMAT=A1,$
FIELDNAME=DISPOSITION, ALIAS=DISPO, FORMAT=A1,$
FIELDNAME=LAB_ORDER, ALIAS=LAB, FORMAT=A1,$
FIELDNAME=PRESCRIPTION, ALIAS=RX, FORMAT=A1,$
FIELDNAME=PURP_VISIT, ALIAS=PRPVST, FORMAT=A1,$
FIELDNAME=JOB_REL_SURV, ALIAS=RELSURV, FORMAT=A1,$
FIELDNAME=ADMIN_REAS, ALIAS=ADMREAS, FORMAT=A1,$
FIELDNAME=DIAG1_RULE, ALIAS=DX1RULE, FORMAT=A1,$
FIELDNAME=DIAG1_CODE, ALIAS=DX1CODE, FORMAT=A5,$

SEGNAME=V_DIAG2, PARENT=LITHO, SEGTYPE=S1,$
FIELDNAME=DIAG2_CODE, ALIAS=DX2CODE, FORMAT=A5,$
FIELDNAME=DIAG2_RULE, ALIAS=DX2RULE, FORMAT=A1,$

SEGNAME=V_PROC, PARENT=LITHO, SEGTYPE=S1,$
FIELDNAME=PROC_CODE, ALIAS=PCODE, FORMAT=A5,$

SEGNAME=M_CODE, PARENT=LITHO, SEGTYPE=S1,$
FIELDNAME=M_CODE, ALIAS=MCODE, FORMAT=A6,$
FIELDNAME=M_DESCR, ALIAS=MDESCR, FORMAT=A32,$

SEGNAME=V_SUPPL, PARENT=LITHO, SEGTYPE=S1,$
FIELDNAME=SUPPL_DISP, ALIAS=SUPPDISP, FORMAT=A1,$

SEGNAME=V_XRAY, PARENT=LITHO, SEGTYPE=S1,$
FIELDNAME=X_RAY, ALIAS=XRAY, FORMAT=A1,$

SEGNAME=V_OTHER, PARENT=LITHO, SEGTYPE=S1,$
FIELDNAME=OTHER_ORDER, ALIAS=OTHORD, FORMAT=A1,$

SEGNAME=V_SPEC, PARENT=LITHO, SEGTYPE=S1,$
FIELDNAME=SPECIAL_PROG, ALIAS=SPROG, FORMAT=A2,$

SEGNAME=V_PSY, PARENT=LITHO, SEGTYPE=S1,$
FIELDNAME=ASSESMNT, ALIAS=ASSESMNT, FORMAT=A2,$
END
DBA=XYZQQ,$

```

The last line of the master file is the Data Base Administrator information. In this case the file is protected by a password given by XYZQQ (not the real password).

Notice the PATIENT_ID field has eight characters. The PC master file has 11 characters. Normally the patient identification is the Social Security number (nine digits) followed by the family member prefix (FMP) (two digits). However, on the mainframe the Social Security number is encrypted. The encryption algorithm reduces the nine digit number to a six character code.

Thus the encrypted Social Security number followed by the FMP results in eight characters.

6.4.1.2 CLINIC MASTER FILE

```
FILENAME=CLINIC,SUFFIX=FOC,$
SEGNAME=CLINIC,SEGTYPE=S1,$
  FIELDNAME=CLINIC,ALIAS=CLUCA,FORMAT=A8,FIELDTYPE=I,$
  FIELDNAME=CL_TITLE,  ALIAS=302TITLE,  FORMAT=A20,$
  FIELDNAME=CL_302CODE, ALIAS=302CLCODE, FORMAT=A2 ,,$
  FIELDNAME=CL_302LINE, ALIAS=302CLLINE, FORMAT=I3 ,,$
END
DBA=XYZQQ,$
```

A brief explanation of this master file is necessary. First notice the format of the CLINIC field. It is eight characters in length. This is because the site identification code, a four-digit number is included as the first part of the UCA code. Thus for BAMC which has a site code of 0109, a UCA code of BALA would look like 0109BALA. Secondly, the third and fourth field names, CL_302CODE and CL_302LINE were designed at the beginning of the study but were never used. Thus these latter fields contain no data.

6.4.1.3 PATIENT MASTER FILE

```
FILENAME=PATIENT,SUFFIX=FOC,$
SEGNAME=PTSEG,SEGTYPE=S1,$
FIELDNAME=PATIENT_ID,ALIAS=PTID,FORMAT=A12,FIELDTYPE=I,$
  FIELDNAME=PT_DOB,  ALIAS=PATDOB,  FORMAT=I6YMD,$
  FIELDNAME=PT_CATEGORY,ALIAS=CATEG,  FORMAT=A3 ,,$
  FIELDNAME=PT_LITHO,  ALIAS=PTLITHO,  FORMAT=A8 ,,$
  FIELDNAME=PT_REGDATE, ALIAS=REGDATE,  FORMAT=I6YMD,$
  FIELDNAME=PT_GENDER,  ALIAS=SEX,  FORMAT=A1 ,,$
  FIELDNAME=PT_RACE,  ALIAS=RACE,  FORMAT=A1 ,,$
  FIELDNAME=PT_PREFIX, ALIAS=PREFIX,  FORMAT=A2 ,,$
  FIELDNAME=PT_PAY_GRADE,ALIAS=PAYGRADE, FORMAT=A2 ,,$
  FIELDNAME=PT_JOB_CODE,ALIAS=JOBCODE,  FORMAT=A4 ,,$
  FIELDNAME=PT_LOCATION,ALIAS=PTLOC,  FORMAT=A6 ,,$
  FIELDNAME=PT_TRN_TDY, ALIAS=TRAINEE,  FORMAT=A1 ,,$
  FIELDNAME=PT_VAELIG,  ALIAS=VA,  FORMAT=A1 ,,$
  FIELDNAME=PT_HCI,  ALIAS=HCI,  FORMAT=A1 ,,$
  FIELDNAME=PT_ZIP_CODE,ALIAS=ZIP,  FORMAT=A5 ,,$
  FIELDNAME=PT_DUAL_SSN,ALIAS=PTDUALSSN, FORMAT=A6 ,,$
  FIELDNAME=PT_FOREIGN, ALIAS=PTFORGN,  FORMAT=A3 ,,$
END
DBA=XYZQQ,$
```

The PATIENT_ID field has 12 characters in it. The

field contains a four-digit site identification code followed by the same eight characters in the visit master file called PATIENT_ID.

6.4.1.4 PROVIDER MASTER FILE

```
FILENAME=PROVIDER,SUFFIX=FOC,$
SEGNAME=PROVIDER,SEGTYPE=S1,$
FIELDNAME=PROV_ID,ALIAS=PROVID,FORMAT=A9,FIELDTYPE=I,$
FIELDNAME=PROV_DATE, ALIAS=PROVDATE, FORMAT=I6YMD,$
FIELDNAME=PROV_STAT, ALIAS=PRSTAT,    FORMAT=A1 ,,$
FIELDNAME=PROV_CATEG, ALIAS=PRCATEG,   FORMAT=A2 ,,$
FIELDNAME=PROV_PRPOS, ALIAS=PRPOS,     FORMAT=A2 ,,$
FIELDNAME=PROV_PAYGR, ALIAS=PRPAYGR,   FORMAT=A2 ,,$
FIELDNAME=PROV_JOBCODE,ALIAS=PRJCODE,  FORMAT=A7 ,,$
FIELDNAME=PROV_LITHO, ALIAS=PRLTH,     FORMAT=A8 ,,$
END
DBA=XYZQQ,$
```

The PROV_ID field, normally the first letter of the last name followed by the last four digits of the provider's Social Security number, is preceded by the four-digit site identification number bringing the format length to nine characters.

6.4.1.5 PROCEDURE MASTER FILE

```
FILENAME=PROCEDUR,SUFFIX=FOC,$
SEGNAME=PROCEDUR,SEGTYPE=S1,$
FIELDNAME=PROC_CODE,ALIAS=PRCCD,FORMAT=A5,FIELDTYPE=I,$
FIELDNAME=PROC_DESCR, ALIAS=PRCDSC,    FORMAT=A67,$
END
DBA=XYZQQ,$
```

Except for the file name, the old and new procedure master files are identical.

6.4.1.6 DIAGNOSIS MASTER FILE

```
FILENAME=DIAGNOS,SUFFIX=FOC,$
SEGNAME=DIAGNOS,SEGTYPE=S1,$
FIELDNAME=DIAG_CODE,ALIAS=DGNCD,FORMAT=A5,FIELDTYPE=I,$
FIELDNAME=DIAG_DESCR, ALIAS=DGNDSC,    FORMAT=A50,$
END
DBA=XYZQQ,$
```

Except for the file name, the old and new diagnoses master files are identical.

6.4.1.7 NEW VISIT MASTER FILE

FILENAME=NBAMC,SUFFIX=FOC,\$

SEGNAME=SITE_ID,SEGTYPE=S1,\$
FIELDNAME=SITE, ALIAS=SID, FORMAT=A4,\$

SEGNAME=VISLOC,PARENT=SITE_ID,SEGTYPE=S1,\$
FIELDNAME=CLINIC, ALIAS=CL_UCA, FORMAT=A4,\$

SEGNAME=VISDTE,PARENT=VISLOC,SEGTYPE=S1,\$
FIELDNAME=VISIT_DATE, ALIAS=VDATE, FORMAT=I6YMD,\$

SEGNAME=VISPROV,PARENT=VISDTE,SEGTYPE=S1,\$
FIELDNAME=PROV_ID, ALIAS=PROV1, FORMAT=A5,\$

SEGNAME=VISITOR,PARENT=VISPROV,SEGTYPE=S1,\$
FIELDNAME=PATIENT_ID, ALIAS=PTID, FORMAT=A8,\$

SEGNAME=LITHO,PARENT=VISITOR,SEGTYPE=S1,\$
FIELDNAME=LITHO_CODE, ALIAS=LITHO, FORMAT=I8,\$
FIELDNAME=FORM_NUM, ALIAS=FNUM, FORMAT=A2,\$
FIELDNAME=VISIT_CNT, ALIAS=VCNT, FORMAT=I1,\$
FIELDNAME=PROV1_TIME, ALIAS=TIME1, FORMAT=I3,\$
FIELDNAME=PROVIDER_2, ALIAS=PROV2, FORMAT=A5,\$
FIELDNAME=PROV2_TIME, ALIAS=TIME2, FORMAT=I3,\$
FIELDNAME=PROV2_RESON, ALIAS=PROV2RES, FORMAT=A1,\$
FIELDNAME=APPT_STATUS, ALIAS=APPSTAT, FORMAT=A1,\$
FIELDNAME=PLACE_OF_VIS, ALIAS=PLACE, FORMAT=A1,\$
FIELDNAME=INP_REFERAL, ALIAS=REFERAL, FORMAT=A4,\$
FIELDNAME=JOB_RELATED, ALIAS=JOBREL, FORMAT=A1,\$
FIELDNAME=MIL_DUTY, ALIAS=DUTY, FORMAT=A1,\$
FIELDNAME=MIL_QTR, ALIAS=QTR, FORMAT=A1,\$
FIELDNAME=MIL_PROF, ALIAS=PROF, FORMAT=A1,\$
FIELDNAME=NOT_AVAIL, ALIAS=NAVAIL, FORMAT=A1,\$
FIELDNAME=ADMITTED, ALIAS=ADMIT, FORMAT=A1,\$
FIELDNAME=INFIELD, ALIAS=ILLN, FORMAT=A1,\$
FIELDNAME=INJURIES, ALIAS=INJ, FORMAT=A1,\$
FIELDNAME=PURP_VIS, ALIAS=PURP, FORMAT=A1,\$
FIELDNAME=RO_F_UP, ALIAS=RULE, FORMAT=A1,\$
FIELDNAME=DIAG1_CODE, ALIAS=DX1CODE, FORMAT=A5,\$

SEGNAME=V_PROC,PARENT=LITHO,SEGTYPE=S1,\$
FIELDNAME=PROC_CODE, ALIAS=PCODE, FORMAT=A5,\$
FIELDNAME=PROV_CD, ALIAS=PCODE, FORMAT=A1,\$

SEGNAME=V_SPCC,PARENT=LITHO,SEGTYPE=S1,\$
FIELDNAME=SP_PRE_CODE, ALIAS=SPCC, FORMAT=A1,\$

SEGNAME=V_OTHER,PARENT=LITHO,SEGTYPE=S1,\$
FIELDNAME=OTHER_CODE, ALIAS=OTHER, FORMAT=A5,\$

```

SEGNAME=V_GROUP, PARENT=LITHO, SEGTYPE=S1, $
  FIELDNAME=UNIT_ID,      ALIAS=UIC,      FORMAT=A6, $
  FIELDNAME=TIME_PR1,     ALIAS=TIMPR1,    FORMAT=I3, $
  FIELDNAME=TIME_PR2,     ALIAS=TIMPR2,    FORMAT=I3, $
  FIELDNAME=TIME_TR1,     ALIAS=TIMTR1,    FORMAT=I3, $
  FIELDNAME=TIME_TR2,     ALIAS=TIMTR2,    FORMAT=I3, $
  FIELDNAME=NO_ACT_DUTY,  ALIAS=MACT,     FORMAT=I3, $
  FIELDNAME=NO_OTH_DUTY,  ALIAS=OACT,     FORMAT=I3, $
  FIELDNAME=NO_RET_MIL,   ALIAS=RMIL,     FORMAT=I3, $
  FIELDNAME=NO_DEPEND,    ALIAS=DEPD,     FORMAT=I3, $
  FIELDNAME=NO_CIVIL,     ALIAS=CIVIL,    FORMAT=I3, $
  FIELDNAME=CONT_SHEET    ALIAS=CSHEET,    FORMAT=A1, $

SEGNAME=V_SPEC, PARENT=LITHO, SEGTYPE=S1, $
  FIELDNAME=SPEC_PROG,    ALIAS=SPROG,     FORMAT=A1, $

SEGNAME=V_DIAG2, PARENT=LITHO, SEGTYPE=S1, $
  FIELDNAME=DIAG2_CODE,   ALIAS=DX2CODE,   FORMAT=A5, $

SEGNAME=V_PSY, PARENT=LITHO, SEGTYPE=S1, $
  FIELDNAME=ASSESMENT,    ALIAS=ASSESMENT, FORMAT=A1, $

SEGNAME=V_DISPOS, PARENT=LITHO, SEGTYPE=S1, $
  FIELDNAME=DISPOS,       ALIAS=DISP,      FORMAT=A2, $

SEGNAME=V_GRSSN, PARENT=LITHO, SEGTYPE=S1, $
  FIELDNAME=GR_PAT_ID,    ALIAS=GPATID,    FORMAT=A11, $
END
DBA=XYZQQ, $

```

6.4.1.8 OTHER CODE MASTER FILE

```

FILENAME=OTHER, SUFFIX=FOC, $

SEGNAME=OTHER, SEGTYPE=S1, $
  FIELDNAME=OTHER_CODE, ALIAS=OTHC, FORMAT=A5, FIELDTYPE=I, $
  FIELDNAME=OTHER_DESCR, ALIAS=OTHDESC, FORMAT=A67, $
END
DBA=XYZQQ, $

```

6.4.2 FIELD DESCRIPTIONS

Except for cases mentioned in the master file section, the field descriptions are the same as in the PC version. See the PC section (4.7) for field sizes and contents.

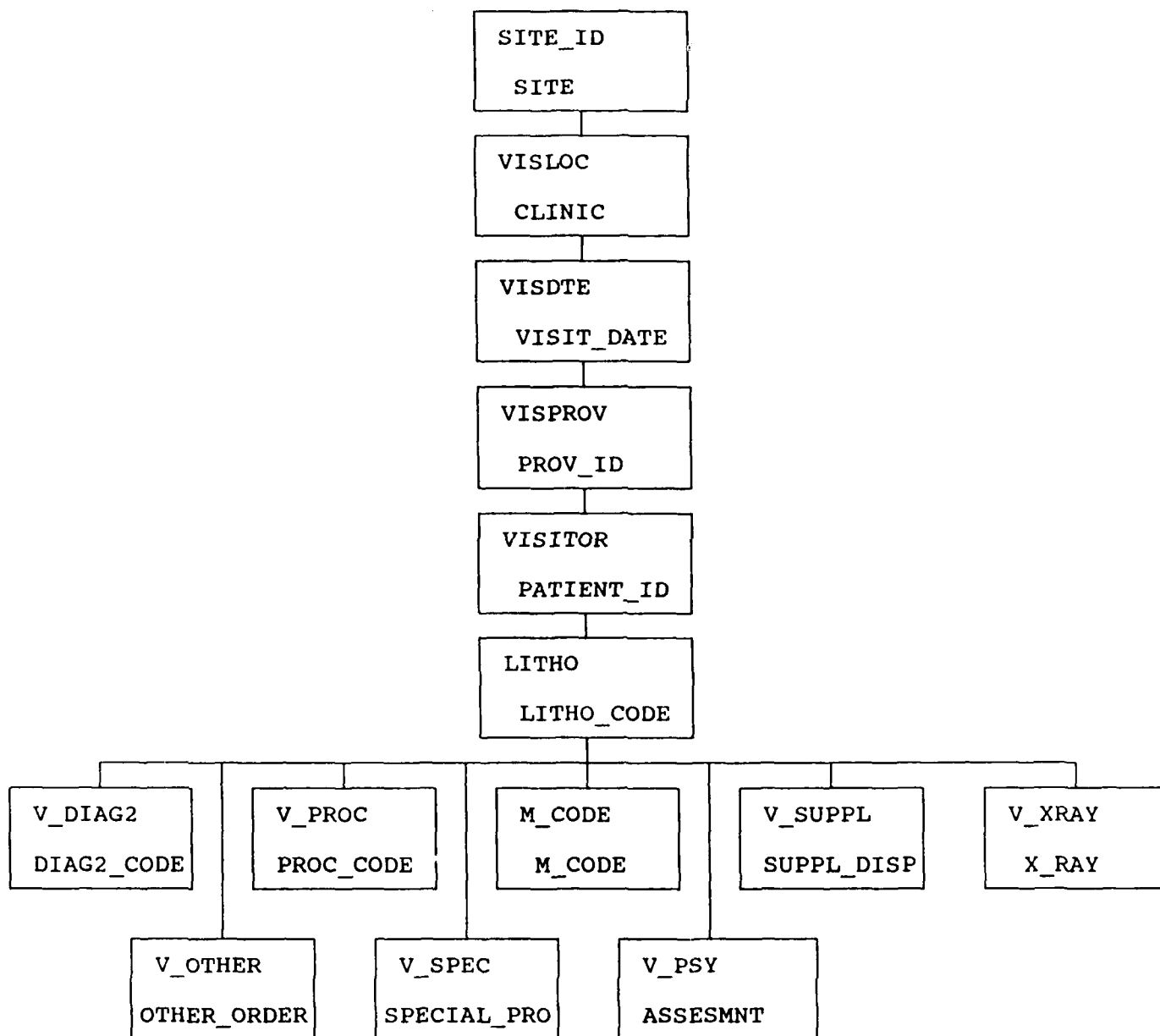
6.4.3 PICTURE OF DATA BASE

All the data files except visit data are single segment. In other words, the hierarchical structure is one level although there are multiple instances of

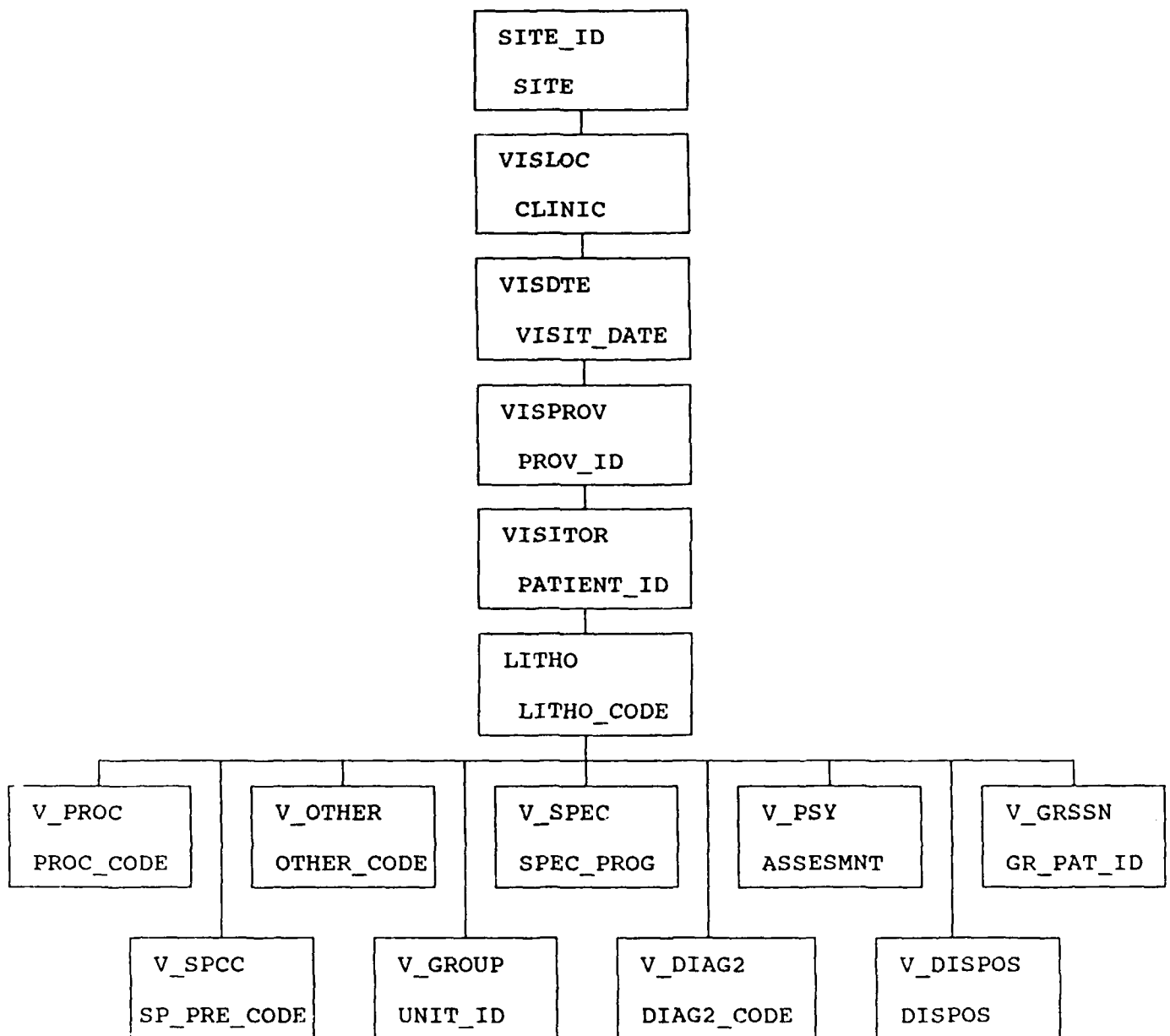
segments. The visit data base has several levels and a picture of the data base helps in understanding the structure. The PC version which is shown elsewhere in the manual is identical except for the top segment which contains the site identification.

A mainframe picture is shown here for completeness. There are two versions, however, one for the old data and one for the new data after the forms redesign of May 1987.

PICTURE DESCRIPTION OF OLD VISIT DATA BASE



PICTURE DESCRIPTION OF NEW VISIT DATA BASE



6.4.4 ENCRYPTION OF THE SSN

Encryption of the Social Security number has been discussed previously. Confidentiality of medical data requires that unauthorized personnel not have access to data and that the medical information not be traced to the patient. The current data base is protected in two ways. The data is password protected and identifying information such as the Social Security number is encrypted. The encryption took place at the site before the data left the hospital. Thus had any disk cartridge fallen into the wrong hands, the data was protected.

An explicit description of the encryption algorithm is not given here because confidentiality of the data base currently residing on the mainframe computer at Ft. Detrick is required. The algorithm does, however, use a 32-base number system which serves two useful purposes. First, for every Social Security number there is a unique encryption. This is necessary since the patient's visit data must be joined or connected to the patient's registration data which is a one-to-one correspondence. Further, if the same patient is seen more than once, the encrypted Social Security numbers must be able to identify these cases. Secondly, using a 32-based number system allows transforming a nine digit number into a six character field. Thus three characters are saved for every patient visit and every patient registered in the data base.

6.5 REFERENCES

FOCUS Users Manual. Release 5.0. (1985). New York, NY: Information Builders, Inc.

Interactive System Productivity Facility/Program Development Facility. Program Reference. SC34-2089-1. (1983). Poughkeepsie, NY: International Business Machines Corporation.

IRMA Domestic User's Manual. (1985). Alpharetta, GA: Digital Communications Associates, Inc.

IRMALink FT/TSO and FT/CMS Users's Manual. (1985). Alpharetta, GA: Digital Communications Associates, Inc.

MVS JCL. GC28-1300-1. (1983). Poughkeepsie, NY: International Business Machines Corporation.

OPTIMIS Operation Management Information System User Manual. (1985). Washington, DC: Headquarters, Department of the Army.

OS/VS2 TSO Command Language Reference. GC28-0646-4.
(1980). Poughkeepsie, NY: International Business
Machines Corporation.

OS/VS2 TSO Terminal User's Guide. GC28-0645-4.
(1980). Poughkeepsie, NY: International Business
Machines Corporation.

SAS User's Guide: Basics. Version 5 Edition. (1985).
Cary, NC: SAS Institute Inc.

DISTRIBUTION:

Deputy Under Secretary (Operations Research), Department of the Army, ATTN:
Mr. Walter Hollis, The Pentagon, Rm 2E660, Wash DC 20310-0200 (1)
Army Study Program Management Office, ATTN: DACS-DMO/Mrs. Joann Langston, The
Pentagon, Rm 3C567, WASH DC 20310-0200 (1)
Deputy Assistant Secretary of Defense (Medical Resources Management), Rm
3E336, The Pentagon, WASH DC 20310-2300 (1)
Resource Analysis & Management System, ATTN: OASD-HA/MAJ S. Baker, 3 Skyline
Place, Suite 1507, 5201 Leesburg Pike, Falls Church, VA 22041-3203 (1)
HQ HSC (HSCL-A), Fort Sam Houston, TX 78234-6000 (1)
Dir, The Army Library, ATTN: ANR-AL-RS (Army Studies), Rm 1A518, The
Pentagon, WASH DC 20310-2300 (1)
Administrator, Defense Logistics Agency, DTIC, ATTN: DTIC-DDAB, Cameron
Station, Alexandria, VA 22304-6145 (2)
Defense Logistics Studies Information Exchange, ALMC, ATTN: Mrs. Alter, Ft
Lee, VA 23801-6043 (1)
Dir, Joint Medical Library, DASG-AAFJML, Offices of The Surgeons General,
Army/Air Force, Rm 670, 5109 Leesburg Pike, Falls Church, VA 22041-3258
HQDA (DASG-HCD-D), 5109 Leesburg Pike, Falls Church, VA 22041-3258 (1)
HQDA (DASG-RMP), 5109 Leesburg Pike, Falls Church, VA 22041-3258 (1)
HQDA (DASG-RMB), 5109 Leesburg Pike, Falls Church, VA 22041-3258 (1)
HQDA (DASG-PSA), 5109 Leesburg Pike, Falls Church, VA 22041-3258 (1)
Medical Library, BAMC, Reid Hall, Bldg 1001, Fort Sam Houston, TX 78234-6200
(1)
Stimson Library, AHS, Bldg. 2840, Fort Sam Houston, TX 78234-6100 (1)

APPENDIX A

SC FILES (REFERENCED SECTION 2.8)

APPENDIX A
RESOLUTION SHEETS
TABLE OF CONTENTS

SCAN400.FIL	Adolescent Medicine
SCAN410.FIL	Allergy
SCAN420.FIL	Audiology/Speech
SCAN430.FIL	Cardiology
SCAN440.FIL	Cardiothoracic
SCAN450.FIL	Dermatology
SCAN460.FIL	Endocrine
SCAN470.FIL	ENT
SCAN480.FIL	Family Practice
SCAN490.FIL	Gastroenterology
SCAN500.FIL	General Surgery
SCAN510.FIL	Gynecology
SCAN520.FIL	Infectious Diseases
SCAN530.FIL	Internal Medicine
SCAN540.FIL	Nephrology/Dialysis
SCAN550.FIL	Neurology
SCAN560.FIL	Neurosurgery
SCAN580.FIL	Nutrition
SCAN590.FIL	OB
SCAN600.FIL	Oncology/Hematology
SCAN610.FIL	Ophthalmology
SCAN620.FIL	Optometry
SCAN630.FIL	Ortho Appliance/Cast
SCAN640.FIL	Orthopedics
SCAN650.FIL	Occupational Therapy
SCAN660.FIL	Pain/Physical Medicine
SCAN670.FIL	Pediatrics
SCAN680.FIL	Physical Therapy
SCAN690.FIL	Plastic Surgery
SCAN700.FIL	Podiatry
SCAN710.FIL	Preventive Medicine
SCAN720.FIL	Primary Care
SCAN730.FIL	Psychiatry
SCAN740.FIL	Psychology
SCAN750.FIL	Pulmonary
SCAN770.FIL	Rheumatology
SCAN780.FIL	Social Work
SCAN790.FIL	Urology
SCAN800.FIL	BAS/TMC
SCAN820.FIL	EKG
SCAN830.FIL	Emergency Room
SCAN840.FIL	Group form I
SCAN850.FIL	Group form II
SCAN860.FIL	Immunization
SCAN900.FIL	Occupational Health

SCAN910.FIL Occupational Therapy Repeat
SCAN920.FIL Physical Therapy Repeat
SCAN930.FIL Repeat
SCAN940.FIL Short
SCAN950.FIL Social Work Short

400 ADOLESCENT MEDICINE

SCAN400.FIL

H02830247NY0051ADOLESCENT MEDICINE

*

H0343

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
B	OTHER UCA 1	217170081800109	V0002000005
B	OTHER UCA 2	2121200818001024	V0002010006
B	DATE (MO)	2434300574601014	V0012020007
B	DATE (DY) 1	1454500555201014	V0004010008
B	DATE (DY) 2	1444400554601014	V0010010009
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014	V0013010011
B	PROV1 CODE	1423901332401014	V0040040012
B	SSN	1332501554601014	V0090090013
B	FMP	1373601554601014	V0020020014
B	VISIT COUNT	2141400544601024	V0009010015
B	PROV1 TIME	2373700351401014	V0022020016
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024	V0013010018
B	PROV2 CODE	1272401332401024	V0040040019
B	PROV2 TIME	2313100351401024	V0022020020
B	BREASON FOR #2 (1)	230300011090109	V0003000021
B	BREASON FOR #2 (2)	2272501080800024	H0003010022
B	IF NOT SCHED	2171700717001024	V0002010023
B	BIF NOT CLINIC (1)	217170063610109	V0003000024
B	BIF NOT CLINIC (2)	2141201606000024	H0003010025
B	REF CODE PFX	2040400706002024	V0006010026
B	REF CODE 1	0030300716001024	V0012010027
B	REF CODE 2	0020101714501024	V0054020028
F	JOB RELATED	801	010029
F	MIL ONLY DUTY	801	010030
F	MIL ONLY QTRS	801	010031
F	MIL ONLY PROF	801	010032
F	FILLER	805	050033
B	SPEC PRE CLIN	306060022140111	H0009090034
B	NOT AVAILABLE	320200009070111	H0003030035
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010036
B	ADDL PROC1	1110701071601024	V0050050037
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010038
B	ADDL PROC2	1050101071601024	V0050050039
B	ADMITTED	211110019190011	V0001010040
B	BUNL REAS PRIM PFX	2121200747602024	V0002010041
B	UNL REAS PRIM	1110701748301034	V0050050042
B	UNL REAS SEC PFX	2060600747602024	V0002010043
B	UNL REAS SEC	1050101748301034	V0050050044
B	FOLLOW/R OUT	2383800272601024	V0002010045
B	PROC PROV1 (A)	347470008090111	H0002020046
B	PROC PROV1 (B)	347470011160111	H0006060047
B	PROC PROV1 (C)	335350008110111	H0004040048
B	PROC PROV1 (D)	323230008090111	H0002020049
B	PROC PROV1 (E)	323230011240111	H0014140050
B	PROC PROV2 (A)	346460008090111	H0002020051

B	PROC	PROV2	(B)	346460011160111	H0006060052
B	PROC	PROV2	(C)	334340008110111	H0004040053
B	PROC	PROV2	(D)	322220008090111	H0002020054
B	PROC	PROV2	(E)	322220011240111	H0014140055
B	DX1	COL1	(A)	247470031460109	V0016000056
B	DX1	COL1	(B)	247470048500109	V0003000057
B	DX1	COL1	(C)	247470052530109	V0002000058
B	DX1	COL1	(D)	247470056600109	V0005000059
B	DX1	COL1	(E)	2474700637001024	V0008020060
B	DX1	COL2	(A)	235350029400109	V0012000061
B	DX1	COL2	(B)	235350043460109	V0004000062
B	DX1	COL2	(C)	235350049550109	V0007000063
B	DX1	COL2	(D)	235350057660109	V0010000064
B	DX1	COL2	(E)	2353500687001024	V0003020065
B	DX1	COL3	(A)	223230029290009	V0001000066
B	DX1	COL3	(B)	223230031330109	V0003000067
B	DX1	COL3	(C)	223230036390109	V0004000068
B	DX1	COL3	(D)	223230042470109	V0006000069
B	DX1	COL3	(E)	223230050500009	V0001000070
B	DX1	COL3	(F)	223230053540109	V0002000071
B	DX1	COL3	(G)	223230056570109	V0002000072
B	DX1	COL3	(H)	2232300596901024	V0011020073
B	DX1	COL4	(A)	211110029350109	V0007000074
B	DX1	COL4	(B)	211110038420109	V0005000075
B	DX1	COL4	(C)	211110044470109	V0004000076
B	DX1	COL4	(D)	211110049520109	V0004000077
B	DX1	COL4	(E)	211110055560109	V0002000078
B	DX1	COL4	(F)	2111100585901024	V0002020079
B	DX2	(A)		346460031460111	H0016160080
B	DX2	(B)		346460048500111	H0003030081
B	DX2	(C)		346460052530111	H0002020082
B	DX2	(D)		346460056600111	H0005050083
B	DX2	(E)		346460063700111	H0008080084
B	DX2	(F)		334340029400111	H0012120085
B	DX2	(G)		334340043460111	H0004040086
B	DX2	(H)		334340049540111	H0006060087
B	DX2	(I)		334340057660111	H0010100088
B	DX2	(J)		334340068700111	H0003030089
B	DX2	(K)		322220029290011	H0001010090
B	DX2	(L)		322220031330111	H0003030091
B	DX2	(M)		322220036390111	H0004040092
B	DX2	(N)		322220042470111	H0006060093
B	DX2	(O)		322220050500011	H0001010094
B	DX2	(P)		322220053540111	H0002020095
B	DX2	(Q)		322220056570111	H0002020096
B	DX2	(R)		322220059690111	H0011110097
B	DX2	(S)		310100029350111	H0007070098
B	DX2	(T)		310100038420111	H0005050099
B	DX2	(U)		310100044470111	H0004040100
B	DX2	(V)		310100049520111	H0004040101
B	DX2	(W)		310100055560111	H0002020102

//EOF0520

410 ALLERGY

SCAN410.FIL

H03830247NY0051ALLERGY

*

H010238

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY) 1	1454500555201014V0004010007	
B	DATE (DY) 2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	223230029290011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800383701024V0002010044	
B	PROC PROV1 (A)	347470008100111	H0003030045
B	PROC PROV1 (B)	347470012130111	H0002020046
B	PROC PROV1 (C)	347470016160011	H0001010047
B	PROC PROV1 (D)	347470019210111	H0003030048
B	PROC PROV1 (E)	347470025250011	H0001010049
B	PROC PROV1 (F)	347470027310111	H0005050050
B	PROC PROV1 (G)	347470033330011	H0001010051

B	PROC	PROV1	(H)	335350009100111	H0002020052
B	PROC	PROV1	(I)	335350012120011	H0001010053
B	PROC	PROV1	(J)	335350014150111	H0002020054
B	PROC	PROV1	(K)	335350017190111	H0003030055
B	PROC	PROV1	(L)	335350021210011	H0001010056
B	PROC	PROV1	(M)	335350024260111	H0003030057
B	PROC	PROV1	(N)	323230008160111	H0009090058
B	PROC	PROV1	(O)	323230019190011	H0001010059
B	PROC	PROV1	(P)	323230021210011	H0001010060
B	PROC	PROV1	(Q)	323230023250111	H0003030061
B	PROC	PROV1	(R)	323230027270011	H0001010062
B	PROC	PROV2	(A)	346460008100111	H0003030063
B	PROC	PROV2	(B)	346460012130111	H0002020064
B	PROC	PROV2	(C)	346460016160011	H0001010065
B	PROC	PROV2	(D)	346460019210111	H0003030066
B	PROC	PROV2	(E)	346460025250011	H0001010067
B	PROC	PROV2	(F)	346460027310111	H0005050068
B	PROC	PROV2	(G)	346460033330011	H0001010069
B	PROC	PROV2	(H)	334340009100111	H0002020070
B	PROC	PROV2	(I)	334340012120011	H0001010071
B	PROC	PROV2	(J)	334340014150111	H0002020072
B	PROC	PROV2	(K)	334340017190111	H0003030073
B	PROC	PROV2	(L)	334340021210011	H0001010074
B	PROC	PROV2	(M)	334340024260111	H0003030075
B	PROC	PROV2	(N)	322220008160111	H0009090076
B	PROC	PROV2	(O)	322220019190011	H0001010077
B	PROC	PROV2	(P)	322220021210011	H0001010078
B	PROC	PROV2	(Q)	322220023250111	H0003030079
B	PROC	PROV2	(R)	322220027270011	H0001010080
B	DX1	COL1	(A)	247470041410009	V0001000081
B	DX1	COL1	(B)	247470043430009	V0001000082
B	DX1	COL1	(C)	2474700476101024	V0015020083
B	DX1	COL2	(A)	235350041410009	V0001000084
B	DX1	COL2	(B)	235350043430009	V0001000085
B	DX1	COL2	(C)	235350045460109	V0002000086
B	DX1	COL2	(D)	235350048600109	V0013000087
B	DX1	COL2	(E)	2353500626301024	V0002020088
B	DX1	COL3	(A)	223230040400009	V0001000089
B	DX1	COL3	(B)	223230042490109	V0008000090
B	DX1	COL3	(C)	223230053590109	V0007000091
B	DX1	COL3	(D)	223230061680109	V0008000092
B	DX1	COL3	(E)	211110040420109	V0003000093
B	DX1	COL3	(F)	211110044440009	V0001000094
B	DX1	COL3	(G)	211110046470109	V0002000095
B	DX1	COL3	(H)	2111100494900024	V0001020096
B	DX2	(A)		346460041410011	H0001010097
B	DX2	(B)		346460043430011	H0001010098
B	DX2	(C)		346460047610111	H0015150099
B	DX2	(D)		334340041410011	H0001010100
B	DX2	(E)		334340043430011	H0001010101
B	DX2	(F)		334340045460111	H0002020102
B	DX2	(G)		334340048600111	H0013130103
B	DX2	(H)		334340062630111	H0002020104
B	DX2	(I)		322220040400011	H0001010105
B	DX2	(J)		322220042490111	H0008080106

B	DX2 (K)	322220053590111	H0007070107
B	DX2 (L)	322220061680111	H0008080108
B	DX2 (M)	310100040420111	H0003030109
B	DX2 (N)	310100044440011	H0001010110
B	DX2 (O)	310100046470111	H0002020111

//EOF0497

H03830247NY0051AUDIOLOGY/SPEECH

*

H010222

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA	2171700818100024H0001010005	
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY) 1	1454500555201014V0004010007	
B	DATE (DY) 2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
F	ADMITTED	801	010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800414001024V0002010044	
B	PROC PROV1 (A)	347470008120111	H0005050045
B	PROC PROV1 (B)	347470014190111	H0006060046
B	PROC PROV1 (C)	347470021240111	H0004040047
B	PROC PROV1 (D)	323230009150111	H0007070048
B	PROC PROV1 (E)	323230018250111	H0008080049
B	PROC PROV1 (F)	323230028370111	H0010100050
B	PROC PROV1 (G)	311110020220111	H0003030051

B	PROC	PROV1	(H)	311110025300111	H0006060052
B	PROC	PROV2	(A)	346460008120111	H0005050053
B	PROC	PROV2	(B)	346460014190111	H0006060054
B	PROC	PROV2	(C)	346460021240111	H0004040055
B	PROC	PROV2	(D)	322220009150111	H0007070056
B	PROC	PROV2	(E)	322220018250111	H0008080057
B	PROC	PROV2	(F)	322220028370111	H0010100058
B	PROC	PROV2	(G)	310100020220111	H0003030059
B	PROC	PROV2	(H)	310100025300111	H0006060060
B	DX1	COL1	(A)	247470046500109	V0005000061
B	DX1	COL1	(B)	247470052570109	V0006000062
B	DX1	COL1	(C)	247470059630109	V0005000063
B	DX1	COL1	(D)	247470065670109	V0003000064
B	DX1	COL1	(E)	247470070730109	V0004000065
B	DX1	COL1	(F)	247470075760109	V0002000066
B	DX1	COL1	(G)	2474700788301024	V0006020067
B	DX1	COL2	(A)	235350046670109	V0022000068
B	DX1	COL2	(B)	235350069690009	V0001000069
B	DX1	COL2	(C)	235350071740109	V0004000070
B	DX1	COL2	(D)	2353500767600024	V0001020071
B	DX1	COL3	(A)	223230044500109	V0007000072
B	DX1	COL3	(B)	223230053550109	V0003000073
B	DX1	COL3	(C)	2232300575700024	V0001020074
B	DX2	(A)		346460046500111	H0005050075
B	DX2	(B)		346460052570111	H0006060076
B	DX2	(C)		346460059630111	H0005050077
B	DX2	(D)		346460065670111	H0003030078
B	DX2	(E)		346460070730111	H0004040079
B	DX2	(F)		346460075760111	H0002020080
B	DX2	(G)		346460078830111	H0006060081
B	DX2	(H)		334340046670111	H0022220082
B	DX2	(I)		334340069690011	H0001010083
B	DX2	(J)		334340071740111	H0004040084
B	DX2	(J1)		334340076760011	H0001010085
B	DX2	(K)		322220044500111	H0007070086
B	DX2	(L)		322220053550111	H0003030087

//EOF0510

H03830247NY0051CARDIOLOGY

*

H010223

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA	2171700818100024H0001010005	
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	223230016160011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800272601024V0002010044	
B	PROC PROV1 (A)	347470008150111	H0008080045
B	PROC PROV1 (B)	347470018190111	H0002020046
B	PROC PROV1 (C)	347470021240111	H0004040047
B	PROC PROV1 (D)	335350008110111	H0004040048
B	PROC PROV1 (E)	335350014150111	H0002020049
B	PROC PROV1 (F)	335350018200111	H0003030050
B	PROC PROV1 (G)	323230008100111	H0003030051

B	PROC PROV1	(H)	323230012140111	H0003030052
B	PROC PROV2	(A)	346460008150111	H0008080053
B	PROC PROV2	(B)	346460018190111	H0002020054
B	PROC PROV2	(C)	346460021240111	H0004040055
B	PROC PROV2	(D)	334340008110111	H0004040056
B	PROC PROV2	(E)	334340014150111	H0002020057
B	PROC PROV2	(F)	334340018200111	H0003030058
B	PROC PROV2	(G)	322220008100111	H0003030059
B	PROC PROV2	(H)	322220012140111	H0003030060
B	DX1 COL1	(A)	247470031330109	V0003000061
B	DX1 COL1	(B)	247470036440109	V0009000062
B	DX1 COL1	(C)	247470047500109	V0004000063
B	DX1 COL1	(D)	247470053530009	V0001000064
B	DX1 COL1	(E)	247470055700109	V0016000065
B	DX1 COL1	(F)	247470072760109	V0005000066
B	DX1 COL1	(G)	2474700798301024	V0005020067
B	DX1 COL2	(A)	235350029340109	V0006000068
B	DX1 COL2	(B)	235350036400109	V0005000069
B	DX1 COL2	(C)	235350043470109	V0005000070
B	DX1 COL2	(D)	235350050530109	V0004000071
B	DX1 COL2	(E)	235350056580109	V0003000072
B	DX1 COL2	(F)	235350061630109	V0003000073
B	DX1 COL2	(G)	235350066690109	V0004000074
B	DX1 COL2	(H)	2353500727501024	V0004020075
B	DX1 COL3	(A)	223230029340109	V0006000076
B	DX1 COL3	(B)	223230036360009	V0001000077
B	DX1 COL3	(C)	223230039470109	V0009000078
B	DX1 COL3	(D)	223230051560109	V0006000079
B	DX1 COL3	(E)	223230058620109	V0005000080
B	DX1 COL3	(F)	2232300646901024	V0006020081
B	DX1 COL4	(A)	211110029320109	V0004000082
B	DX1 COL4	(B)	211110034380109	V0005000083
B	DX1 COL4	(C)	211110040440109	V0005000084
B	DX1 COL4	(D)	211110047530109	V0007000085
B	DX1 COL4	(E)	211110055590109	V0005000086
B	DX1 COL4	(F)	211110062640109	V0003000087
B	DX1 COL4	(G)	2111100666600024	V0001020088
B	DX2	(A)	346460031330111	H0003030089
B	DX2	(B)	346460036440111	H0009090090
B	DX2	(C)	346460047500111	H0004040091
B	DX2	(D)	346460053530011	H0001010092
B	DX2	(E)	346460055700111	H0016160093
B	DX2	(F)	346460072760111	H0005050094
B	DX2	(G)	346460079830111	H0005050095
B	DX2	(H)	334340029340111	H0006060096
B	DX2	(I)	334340036400111	H0005050097
B	DX2	(J)	334340043470111	H0005050098
B	DX2	(K)	334340050530111	H0004040099
B	DX2	(L)	334340056580111	H0003030100
B	DX2	(M)	334340061630111	H0003030101
B	DX2	(N)	334340066690111	H0004040102
B	DX2	(O)	334340072750111	H0004040103
B	DX2	(P)	322220029340111	H0006060104
B	DX2	(Q)	322220036360011	H0001010105
B	DX2	(R)	322220039470111	H0009090106

B	DX2 (S)	322220051560111	H0006060107
B	DX2 (T)	322220058620111	H0005050108
B	DX2 (U)	322220064690111	H0006060109
B	DX2 (V)	310100029320111	H0004040110
B	DX2 (W)	310100034380111	H0005050111
B	DX2 (X)	310100040440111	H0005050112
B	DX2 (Y)	310100047530111	H0007070113
B	DX2 (Z)	310100055590111	H0005050114
B	DX2 (AA)	310100062640111	H0003030115

//EOF0540

440 CARDIOTHERAPY

SCAN440.FIL

H03830247NY0051CARDIOTHERAPY

*

H010218

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	247470023230011	V0001010039
	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800272601024V0002010044	
B	PROC PROV1	347470007210111	H0015150045
B	PROC PROV2	346460007210111	H0015150046
B	DX1 COL1 (A) _	247470032350109	V0004000047
B	DX1 COL1 (B) _	247470037420109	V0006000048
B	DX1 COL1 (C) _	247470045480109	V0004000049
B	DX1 COL1 (D) _	247470051580109	V0008000050
B	DX1 COL1 (E) _	2474700617501024V0015020051	

B	DX1 COL2 (A)	235350029350109	V0007000052
B	DX1 COL2 (B)	235350038390109	V0002000053
B	DX1 COL2 (C)	235350042470109	V0006000054
B	DX1 COL2 (D)	235350050530109	V0004000055
B	DX1 COL2 (E)	235350056590109	V0004000056
B	DX1 COL2 (F)	2353500626601024	V0005020057
B	DX1 COL3 (A)	223230029410109	V0013000058
B	DX1 COL3 (B)	223230044500109	V0007000059
B	DX1 COL3 (C)	2232300536001024	V0008020060
B	DX1 COL4 (A)	211110029370109	V0009000061
B	DX1 COL4 (B)	211110040430109	V0004000062
B	DX1 COL4 (C)	211110046530109	V0008000063
B	DX1 COL4 (D)	211110056590109	V0004000064
B	DX1 COL4 (E)	2111100626301024	V0002020065
B	DX2 (A)	346460032350111	H0004040066
B	DX2 (B)	346460037420111	H0006060067
B	DX2 (C)	346460045480111	H0004040068
B	DX2 (D)	346460051580111	H0008080069
B	DX2 (E)	346460061750111	H0015150070
B	DX2 (F)	334340029350111	H0007070071
B	DX2 (G)	334340038390111	H0002020072
B	DX2 (H)	334340042470111	H0006060073
B	DX2 (I)	334340050530111	H0004040074
B	DX2 (J)	334340056590111	H0004040075
B	DX2 (K)	334340062660111	H0005050076
B	DX2 (L)	322220029410111	H0013130077
B	DX2 (M)	322220044500111	H0007070078
B	DX2 (N)	322220053600111	H0008080079
B	DX2 (O)	310100029370111	H0009090080
B	DX2 (P)	310100040430111	H0004040081
B	DX2 (Q)	310100046530111	H0008080082
B	DX2 (R)	310100056590111	H0004040083
B	DX2 (S)	310100062630111	H0002020084

//EOF0495

450 DERMATOLOGY

SCAN450.FIL

H03830247NY0051DERMATOLOGY

*

H010228

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014	V0012020006
B	DATE (DY)1	1454500555201014	V0004010007
B	DATE (DY)2	1444400554601014	V0010010008
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014	V0013010010
B	PROV1 CODE	1423901332401014	V0040040011
B	SSN	1332501554601014	V0090090012
B	FMP	1373601554601014	V0020020013
B	VISIT COUNT	2141400544601024	V0009010014
B	PROV1 TIME	2373700351401014	V0022020015
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024	V0013010017
B	PROV2 CODE	1272401332401024	V0040040018
B	PROV2 TIME	2313100351401024	V0022020019
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024	H0003010021
B	IF NOT SCHED	2171700717001024	V0002010022
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024	H0003010024
B	REF CODE PFX	2040400706002024	V0006010025
B	REF CODE 1	0030300716001024	V0012010026
B	REF CODE 2	0020101714501024	V0054020027
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024	V0002010029
B	MIL ONLY QTRS	2191900222001024	V0003010030
B	MIL ONLY PROF	2191900181501024	V0004010031
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024	V0050050036
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024	V0050050038
B	ADMITTED	211110036360011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024	V0002010040
B	UNL REAS PRIM	1110701748301034	V0050050041
B	UNL REAS SEC PFX	2060600747602024	V0002010042
B	UNL REAS SEC	1050101748301034	V0050050043
B	FOLLOW/R OUT	2383800393801024	V0002010044
B	BPROC PROV1 (A)	347470008130111	H0006060045
B	BPROC PROV1 (B)	347470015190111	H0005050046
B	BPROC PROV1 (C)	347470021260111	H0006060047
B	BPROC PROV1 (D)	347470028300111	H0003030048
B	BPROC PROV1 (E)	347470032360111	H0005050049
B	BPROC PROV1 (F)	335350008090111	H0002020050
B	BPROC PROV1 (G)	335350011160111	H0006060051

BPROC PROV1 (H)	335350018200111	H0003030052
BPROC PROV1 (I)	335350022260111	H0005050053
BPROC PROV1 (J)	335350028310111	H0004040054
BPROC PROV1 (K)	335350033360111	H0004040055
BPROC PROV1 (L)	323230007110111	H0005050056
BPROC PROV1 (M)	323230013210111	H0009090057
BPROC PROV1 (N)	323230023250111	H0003030058
BPROC PROV1 (O)	323230027320111	H0006060059
BPROC PROV1 (P)	323230034360111	H0003030060
BPROC PROV1 (Q)	311110019350111	H0017170061
BPROC PROV2 (A)	346460008130111	H0006060062
BPROC PROV2 (B)	346460015190111	H0005050063
BPROC PROV2 (C)	346460021260111	H0006060064
BPROC PROV2 (D)	346460028300111	H0003030065
BPROC PROV2 (E)	346460032360111	H0005050066
BPROC PROV2 (F)	334340008090111	H0002020067
BPROC PROV2 (G)	334340011160111	H0006060068
BPROC PROV2 (H)	334340018200111	H0003030069
BPROC PROV2 (I)	334340022260111	H0005050070
BPROC PROV2 (J)	334340028310111	H0004040071
BPROC PROV2 (K)	334340033360111	H0004040072
BPROC PROV2 (L)	322220007110111	H0005050073
BPROC PROV2 (M)	322220013210111	H0009090074
BPROC PROV2 (N)	322220023250111	H0003030075
BPROC PROV2 (O)	322220027320111	H0006060076
BPROC PROV2 (P)	322220034360111	H0003030077
BPROC PROV2 (Q)	310100019350111	H0017170078
B DX1 COL1	2474700428301024	V0042020079
B DX1 COL2	2353500408201024	V0043020080
B DX1 COL3	2232300408301024	V0044020081
B DX1 COL4 (A)	211110040660109	V0027000082
B DX1 COL4 (B)	2111100686800024	V0001020083
B DX2 (A)	346460042830111	H0042420084
B DX2 (B)	334340040820111	H0043430085
B DX2 (C)	322220040830111	H0044440086
B DX2 (D)	310100040660111	H0027270087

//EOF0685

460 ENDOCRINE

H03830247NY0051ENDOCRINE

*

H010242

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	247470019190011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/ R OUT	2383800272601024V0002010044	
B	PROC PROV1	347470007160111	H0010100045
B	PROC PROV2	346460007160111	H0010100046
B	BDX1 COL1 (A)	247470031380109	V0008000047
B	BDX1 COL1 (B)	2474700415001024V0010020048	
B	BDX1 COL2 (A)	235350029340109	V0006000049
B	BDX1 COL2 (B)	235350037400109	V0004000050
B	BDX1 COL2 (C)	2353500435001024V0008020051	

BDX1 COL3 (A)	223230029350109	V0007000052
BDX1 COL3 (B)	223230038470109	V0010000053
BDX1 COL3 (C)	211110029300109	V0002000054
BDX1 COL3 (D)	211110032350109	V0004000055
BDX1 COL3 (E)	211110037370009	V0001000056
BDX1 COL3 (F)	211110039390009	V0001000057
BDX1 COL3 (G)	2111100414100024	V0001020058
B DX2 (A)	346460031380111	H0008080059
B DX2 (B)	346460041500111	H0010100060
B DX2 (C)	334340029340111	H0006060061
B DX2 (D)	334340037400111	H0004040062
B DX2 (E)	334340043500111	H0008080063
B DX2 (F)	322220029350111	H0007070064
B DX2 (G)	322220038470111	H0010100065
B DX2 (H)	310100029300111	H0002020066
B DX2 (I)	310100032350111	H0004040067
B DX2 (J)	310100037370011	H0001010068
B DX2 (K)	310100039390011	H0001010069

//EOF0426

470 ENT

SCAN470.FIL

H03830247NY0051ENT

*

H010243

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014	V0012020006
B	DATE (DY) 1	1454500555201014	V0004010007
B	DATE (DY) 2	1444400554601014	V0010010008
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014	V0013010010
B	PROV1 CODE	1423901332401014	V0040040011
B	SSN	1332501554601014	V0090090012
B	FMP	1373601554601014	V0020020013
B	VISIT COUNT	2141400544601024	V0009010014
B	PROV1 TIME	2373700351401014	V0022020015
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024	V0013010017
B	PROV CODE	1272401332401024	V0040040018
B	PROV2 TIME	2313100351401024	V0022020019
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024	H0003010021
B	IF NOT SCHED	2171700717001024	V0002010022
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024	H0003010024
B	REF CODE PFX	2040400706002024	V0006010025
B	REF CODE 1	0030300716001024	V0012010026
B	REF CODE 2	0020101714501024	V0054020027
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024	V0002010029
B	MIL ONLY QTRS	2191900222001024	V0003010030
B	MIL ONLY PROF	2191900181501024	V0004010031
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024	V0050050036
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024	V0050050038
B	ADMITTED	223230022220011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024	V0002010040
B	UNL REAS PRIM	1110701748301034	V0050050041
B	UNL REAS SEC PFX	2060600747602024	V0002010042
B	UNL REAS SEC	1050101748301034	V0050050043
B	FOLLOW/R OUT	2383800363501024	V0002010044
B	PROC PROV1 (A)	347470008160111	H0009090045
B	PROC PROV1 (B)	347470019300111	H0012120046
B	PROC PROV1 (C)	335350008090111	H0002020047
B	PROC PROV1 (D)	335350011150111	H0005050048
B	PROC PROV1 (E)	335350017170011	H0001010049
B	PROC PROV1 (F)	323230008180111	H0011110050
B	PROC PROV1 (G)	323230020200011	H0001010051

B	PROC	PROV2	(A)	346460008160111	H0009090052
B	PROC	PROV2	(B)	346460019300111	H0012120053
B	PROC	PROV2	(C)	334340008090111	H0002020054
B	PROC	PROV2	(D)	334340011150111	H0005050055
B	PROC	PROV2	(E)	334340017170011	H0001010056
B	PROC	PROV2	(F)	322220008180111	H0011110057
B	PROC	PROV2	(G)	322220020200011	H0001010058
B	DX1	COL1	(A)	247470040430109	V0004000059
B	DX1	COL1	(B)	247470045490109	V0005000060
B	DX1	COL1	(C)	247470051560109	V0006000061
B	DX1	COL1	(D)	2474700586701024	V0010020062
B	DX1	COL2	(A)	235350038450109	V0008000063
B	DX1	COL2	(B)	235350047500109	V0004000064
B	DX1	COL2	(C)	235350052570109	V0006000065
B	DX1	COL2	(D)	2353500607801024	V0019020066
B	DX1	COL3	(A)	223230038390109	V0002000067
B	DX1	COL3	(B)	223230043580109	V0016000068
B	DX1	COL3	(C)	2232300616201024	V0002020069
B	DX1	COL4	(A)	211110038400109	V0003000070
B	DX1	COL4	(B)	211110042420009	V0001000071
B	DX1	COL4	(C)	211110044470109	V0004000072
B	DX1	COL4	(D)	211110049550109	V0007000073
B	DX1	COL4	(E)	2111100575700024	V0001020074
B	DX2	(A)		346460040430111	H0004040075
B	DX2	(A1)		346460045490111	H0005050076
B	DX2	(B)		346460051560111	H0006060077
B	DX2	(C)		346460058670111	H0010100078
B	DX2	(D)		334340038450111	H0008080079
B	DX2	(E)		334340047500111	H0004040080
B	DX2	(F)		334340052570111	H0006060081
B	DX2	(G)		334340060780111	H0019190082
B	DX2	(H)		322220038390111	H0002020083
B	DX2	(I)		322220043580111	H0016160084
B	DX2	(J)		322220061620111	H0002020085
B	DX2	(K)		310100038400111	H0003030086
B	DX2	(L)		310100042420011	H0001010087
B	DX2	(M)		310100044470111	H0004040088
B	DX2	(N)		310100049550111	H0007070089

//EOF0524

H03830247NY0051FAMILY PRACTICE

*

H010237

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
B	OTHER UCA	2171700818100024	V0001010005
B	DATE (MO)	2434300574601014	V0012020006
B	DATE (DY) 1	1454500555201014	V0004010007
B	DATE (DY) 2	1444400554601014	V0010010008
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014	V0013010010
B	PROV1 CODE	1423901332401014	V0040040011
B	SSN	1332501554601014	V0090090012
B	FMP	1373601554601014	V0020020013
B	VISIT COUNT	2141400544601024	V0009010014
B	PROV1 TIME	2373700351401014	V0022020015
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024	V0013010017
B	PROV2 CODE	1272401332401024	V0040040018
B	PROV2 TIME	2313100351401024	V0022020019
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024	H0003010021
B	IF NOT SCHED	2171700717001024	V0002010022
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024	H0003010024
B	REF CODE PFX	2040400706002024	V0006010025
B	REF CODE 1	0030300716001024	V0012010026
B	REF CODE 2	0020101714501024	V0054020027
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024	V0002010029
B	MIL ONLY QTRS	2191900222001024	V0003010030
B	MIL ONLY PROF	2191900181501024	V0004010031
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024	V0050050036
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024	V0050050038
B	ADMITTED	235350014140011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024	V0002010040
B	UNL REAS PRIM	1110701748301034	V0050050041
B	UNL REAS SEC PFX	2060600747602024	V0002010042
B	UNL REAS SEC	1050101748301034	V0050050043
B	FOLLOW/R OUT	2383800222101024	V0002010044
B	PROC PROV1 (A)	347470007180111	H0012120045
B	PROC PROV1 (B)	335350008100111	H0003030046
B	PROC PROV1 (C)	335350012120011	H0001010047
B	PROC PROV2 (A)	346460007180111	H0012120048
B	PROC PROV2 (B)	334340008100111	H0003030049
B	PROC PROV2 (C)	334340012120011	H0001010050
B	DX1 COL1 (A)	247470026320109	V0007000051

B	DX1 COL1 (B)	247470034390109	V0006000052
B	DX1 COL1 (C)	247470043540109	V0012000053
B	DX1 COL1 (D)	247470056630109	V0008000054
B	DX1 COL1 (E)	247470065710109	V0007000055
B	DX1 COL1 (F)	247470073740109	V0002000056
B	DX1 COL1 (G)	2474700768101024	V0006020057
B	DX1 COL2 (A)	235350024330109	V0010000058
B	DX1 COL2 (B)	235350036570109	V0022000059
B	DX1 COL2 (C)	235350060600009	V0001000060
B	DX1 COL2 (D)	2353500637001024	V0008020061
B	DX1 COL3 (A)	223230024460109	V0023000062
B	DX1 COL3 (B)	223230049560109	V0008000063
B	DX1 COL3 (C)	223230059620109	V0004000064
B	DX1 COL3 (D)	223230065660109	V0002000065
B	DX1 COL3 (E)	223230071720109	V0002000066
B	DX1 COL3 (F)	223230074740009	V0001000067
B	DX1 COL3 (G)	223230076760009	V0001000068
B	DX1 COL3 (H)	2232300787800024	V0001020069
B	DX1 COL4 (A)	211110025310109	V0007000070
B	DX1 COL4 (B)	211110033340109	V0002000071
B	DX1 COL4 (C)	211110036360009	V0001000072
B	DX1 COL4 (D)	211110038430109	V0006000073
B	DX1 COL4 (E)	211110046460009	V0001000074
B	DX1 COL4 (F)	211110048480009	V0001000075
B	DX1 COL4 (G)	211110050510109	V0002000076
B	DX1 COL4 (H)	211110054540009	V0001000077
B	DX1 COL4 (I)	211110056590109	V0004000078
B	DX1 COL4 (J)	211110062620009	V0001000079
B	DX1 COL4 (K)	2111100646501024	V0002020080
B	DX2 (A)	346460026320111	H0007070081
B	DX2 (B)	346460034390111	H0006060082
B	DX2 (C)	346460043540111	H0012120083
B	DX2 (D)	346460056630111	H0008080084
B	DX2 (E)	346460065710111	H0007070085
B	DX2 (F)	346460073740111	H0002020086
B	DX2 (G)	346460076810111	H0006060087
B	DX2 (H)	334340024330111	H0010100088
B	DX2 (I)	334340036570111	H0022220089
B	DX2 (J)	334340060600011	H0001010090
B	DX2 (K)	334340063700111	H0008080091
B	DX2 (L)	322220024460111	H0023230092
B	DX2 (M)	322220049560111	H0008080093
B	DX2 (N)	322220059620111	H0004040094
B	DX2 (O)	322220065660111	H0002020095
B	DX2 (P)	322220071720111	H0002020096
B	DX2 (Q)	322220074740011	H0001010097
B	DX2 (R)	322220076760011	H0001010098
B	DX2 (S)	322220078780011	H0001010099
B	DX2 (T)	310100025310111	H0007070100
B	DX2 (U)	310100033340111	H0002020101
B	DX2 (V)	310100036360011	H0001010102
B	DX2 (W)	310100038430111	H0006060103
B	DX2 (X)	310100046460011	H0001010104
B	DX2 (Y)	310100048480011	H0001010105
B	DX2 (Z)	310100050510111	H0002020106

B	DX2 (AA)	310100054540011	H0001010107
B	DX2 (BB)	310100056590111	H0004040108
B	DX2 (CC)	310100062620011	H0001010109

//EOF0534

490 GASTROENTEROLOGY

SCAN490.FIL

H03830247NY0051GASTROENTEROLOGY

*

H010235

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
B	OTHER UCA	2171700818100024	H0001010005
B	DATE (MO)	2434300574601014	V0012020006
B	DATE (DY) 1	1454500555201014	V0004010007
B	DATE (DY) 2	1444400554601014	V0010010008
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014	V0013010010
B	PROV1 CODE	1423901332401014	V0040040011
B	SSN	1332501554601014	V0090090012
B	FMP	1373601554601014	V0020020013
B	VISIT COUNT	2141400544601024	V0009010014
B	PROV1 TIME	2373700351401014	V0022020015
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024	V0013010017
B	PROV2 CODE	1272401332401024	V0040040018
B	PROV2 TIME	2313100351401024	V0022020019
BREASON FOR #2 (1)		230300011090109	V0003000020
BREASON FOR #2 (2)		2272501080800024	H0003010021
B	IF NOT SCHED	2171700717001024	V0002010022
BIF NOT CLINIC (1)		217170063610109	V0003000023
BIF NOT CLINIC (2)		2141201606000024	H0003010024
B	REF CODE PFX	2040400706002024	V0006010025
B	REF CODE 1	0030300716001024	V0012010026
B	REF CODE 2	0020101714501024	V0054020027
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024	V0002010029
B	MIL ONLY QTRS	2191900222001024	V0003010030
B	MIL ONLY PROF	2191900181501024	V0004010031
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
BPROV2 ADDL1 (YES)		208080006060011	V0001010035
B	ADDL PROC1	1110701071601024	V0050050036
BPROV2 ADDL2 (YES)		202020006060011	V0001010037
B	ADDL PROC2	1050101071601024	V0050050038
B	ADMITTED	223230019190011	V0001010039
BUNL REAS PRIM PFX		2121200747602024	V0002010040
B	UNL REAS PRIM	1110701748301034	V0050050041
B	UNL REAS SEC PFX	2060600747602024	V0002010042
B	UNL REAS SEC	1050101748301034	V0050050043
B	FOLLOW/R OUT	2383800272601024	V0002010044
B	PROC PROV1 (A)	347470007070011	H0001010045
B	PROC PROV1 (B)	347470010160111	H0007070046
B	PROC PROV1 (C)	347470018210111	H0004040047
B	PROC PROV1 (D)	335350008100111	H0003030048
B	PROC PROV1 (E)	335350012130111	H0002020049
B	PROC PROV1 (F)	335350015190111	H0005050050
B	PROC PROV1 (G)	323230008100111	H0003030051

B	PROC	PROV1	(H)	323230012140111	H0003030052
B	PROC	PROV1	(I)	323230016160011	H0001010053
B	PROC	PROV2	(A)	346460007070011	H0001010054
B	PROC	PROV2	(B)	346460010160111	H0007070055
B	PROC	PROV2	(C)	346460018210111	H0004040056
B	PROC	PROV2	(D)	334340008100111	H0003030057
B	PROC	PROV2	(E)	334340012130111	H0002020058
B	PROC	PROV2	(F)	334340015190111	H0005050059
B	PROC	PROV2	(G)	322220008100111	H0003030060
B	PROC	PROV2	(H)	322220012140111	H0003030061
B	PROC	PROV2	(I)	322220016160011	H0001010062
B	DX1	COL1	(A)	247470031340109	V0004000063
B	DX1	COL1	(B)	247470036400109	V0005000064
B	DX1	COL1	(C)	247470042430109	V0002000065
B	DX1	COL1	(D)	247470045490109	V0005000066
B	DX1	COL1	(E)	247470052630109	V0012000067
B	DX1	COL1	(F)	2474700656901024	V0005020068
B	DX1	COL2	(A)	235350029380109	V0010000069
B	DX1	COL2	(B)	235350041450109	V0005000070
B	DX1	COL2	(C)	235350047630109	V0017000071
B	DX1	COL2	(D)	235350067690109	V0003000072
B	DX1	COL2	(E)	2353500717501024	V0005020073
B	DX1	COL3	(A)	223230029320109	V0004000074
B	DX1	COL3	(B)	223230034350109	V0002000075
B	DX1	COL3	(C)	223230037390109	V0003000076
B	DX1	COL3	(D)	223230042440109	V0003000077
B	DX1	COL3	(E)	223230046540109	V0009000078
B	DX1	COL3	(F)	223230056680109	V0013000079
B	DX1	COL3	(G)	223230070730109	V0004000080
B	DX1	COL3	(H)	2232300757500024	V0001020081
B	DX2		(A)	346460031340111	H0004040082
B	DX2		(B)	346460036400111	H0005050083
B	DX2		(C)	346460042430111	H0002020084
B	DX2		(D)	346460045490111	H0005050085
B	DX2		(E)	346460052630111	H0012120086
B	DX2		(F)	346460065690111	H0005050087
B	DX2		(G)	334340029380111	H0010100088
B	DX2		(H)	334340041450111	H0005050089
B	DX2		(I)	334340047630111	H0017170090
B	DX2		(J)	334340067690111	H0003030091
B	DX2		(K)	334340071750111	H0005050092
B	DX2		(L)	322220029320111	H0004040093
B	DX2		(M)	322220034350111	H0002020094
B	DX2		(N)	322220037390111	H0003030095
B	DX2		(O)	322220042440111	H0003030096
B	DX2		(P)	322220046540111	H0009090097
B	DX2		(Q)	322220056680111	H0013130098
B	DX2		(R)	322220070730111	H0004040099

//EOF0512

500 GENERAL SURGERY

SCAN500.FIL

H03830247NY0051GENERAL SURGERY

*

H010244

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	223230023230011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800272601024V0002010044	
B	PROC PROV1 (A)	347470008180111	H0011110045
B	PROC PROV1 (B)	347470021240111	H0004040046
B	PROC PROV1 (C)	335350009100111	H0002020047
B	PROC PROV1 (D)	335350013140111	H0002020048
B	PROC PROV1 (E)	335350017210111	H0005050049
B	PROC PROV1 (F)	323230008110111	H0004040050
B	PROC PROV1 (G)	323230014210111	H0008080051

B	PROC	PROV2	(A)	346460008180111	H0011110052
B	PROC	PROV2	(B)	346460021240111	H0004040053
B	PROC	PROV2	(C)	334340009100111	H0002020054
B	PROC	PROV2	(D)	334340013140111	H0002020055
B	PROC	PROV2	(E)	334340017210111	H0005050056
B	PROC	PROV2	(F)	322220008110111	H0004040057
B	PROC	PROV2	(G)	322220014210111	H0008080058
B	DX1	COL1	(A)	247470031430109	V0013000059
B	DX1	COL1	(B)	247470046520109	V0007000060
B	DX1	COL1	(C)	2474700566101024	V0006020061
B	DX1	COL2	(A)	235350029300109	V0002000062
B	DX1	COL2	(B)	235350032340109	V0003000063
B	DX1	COL2	(C)	235350036420109	V0007000064
B	DX1	COL2	(D)	235350045500109	V0006000065
B	DX1	COL2	(E)	235350053540109	V0002000066
B	DX1	COL2	(F)	2353500567001024	V0015020067
B	DX1	COL3	(A)	223230030370109	V0008000068
B	DX1	COL3	(B)	223230039410109	V0003000069
B	DX1	COL3	(C)	223230043480109	V0006000070
B	DX1	COL3	(D)	223230050560109	V0007000071
B	DX1	COL3	(E)	223230059620109	V0004000072
B	DX1	COL3	(F)	223230065670109	V0003000073
B	DX1	COL3	(G)	223230069700109	V0002000074
B	DX1	COL3	(H)	223230072730109	V0002000075
B	DX1	COL3	(I)	2232300758101024	V0007020076
B	DX1	COL4	(A)	211110029310109	V0003000077
B	DX1	COL4	(B)	211110033410109	V0009000078
B	DX1	COL4	(C)	211110043470109	V0005000079
B	DX1	COL4	(D)	211110049500109	V0002000080
B	DX1	COL4	(E)	211110052520009	V0001000081
B	DX1	COL4	(F)	211110054540009	V0001000082
B	DX1	COL4	(G)	2111100565600024	V0001020083
B	DX2		(A)	346460031430111	H0013130084
B	DX2		(B)	346460046520111	H0007070085
B	DX2		(C)	346460056610111	H0006060086
B	DX2		(D)	334340029300111	H0002020087
B	DX2		(E)	334340032340111	H0003030088
B	DX2		(F)	334340036420111	H0007070089
B	DX2		(G)	334340045500111	H0006060090
B	DX2		(H)	334340053540111	H0002020091
B	DX2		(I)	334340056700111	H0015150092
B	DX2		(J)	322220030370111	H0008080093
B	DX2		(K)	322220039410111	H0003030094
B	DX2		(L)	322220043480111	H0006060095
B	DX2		(M)	322220050560111	H0007070096
B	DX2		(N)	322220059620111	H0004040097
B	DX2		(O)	322220065670111	H0003030098
B	DX2		(P)	322220069700111	H0002020099
B	DX2		(Q)	322220072730111	H0002020100
B	DX2		(R)	322220075810111	H0007070101
B	DX2		(S)	310100029310111	H0003030102
B	DX2		(T)	310100033410111	H0009090103
B	DX2		(U)	310100043470111	H0005050104
B	DX2		(V)	310100049500111	H0002020105
B	DX2		(W)	310100052520011	H0001010106

B DX2 (X)
//EOF0541

310100054540011 H0001010107

510 GYNECOLOGY

SCAN510.FIL

H03830247NY0051GYNECOLOGY

*

H010225

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
B	OTHER UCA 1	217170081800109	V0002000005
B	OTHER UCA 2	2121200818100024	V0001010006
B	DATE (MO)	2434300574601014	V0012020007
B	DATE (DY)1	1454500555201014	V0004010008
B	DATE (DY)2	1444400554601014	V0010010009
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014	V0013010011
B	PROV1 CODE	1423901332401014	V0040040012
B	SSN	1332501554601014	V0090090013
B	FMP	1373601554601014	V0020020014
B	VISIT COUNT	2141400544601024	V0009010015
B	PROV1 TIME	2373700351401014	V0022020016
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024	V0013010018
B	PROV2 CODE	1272401332401024	V0040040019
B	PROV2 TIME	2313100351401024	V0022020020
B	BREASON FOR #2 (1)	230300011090109	V0003000021
B	BREASON FOR #2 (2)	2272501080800024	H0003010022
B	IF NOT SCHED	2171700717001024	V0002010023
B	BIF NOT CLINIC (1)	217170063610109	V0003000024
B	BIF NOT CLINIC (2)	2141201606000024	H0003010025
B	REF CODE PFX	2040400706002024	V0006010026
B	REF CODE 1	0030300716001024	V0012010027
B	REF CODE 2	0020101714501024	V0054020028
B	JOB RELATED	217170032320011	V0001010029
B	MIL ONLY DUTY	2202000241410024	V0002010030
B	MIL ONLY QTRS	2191900222001024	V0003010031
B	MIL ONLY PROF	2191900181501024	V0004010032
F	FILLER	805	050033
B	SPEC PRE CLIN	306060022140111	H0009090034
B	NOT AVAILABLE	320200009070111	H0003030035
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010036
B	ADDL PROC1	1110701071601024	V0050050037
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010038
B	ADDL PROC2	1050101071601024	V0050050039
B	ADMITTED	211110024240011	V0001010040
B	BUNL REAS PRIM PFX	2121200747602024	V0002010041
B	UNL REAS PRIM	1110701748301034	V0050050042
B	UNL REAS SEC PFX	2060600747602024	V0002010043
B	UNL REAS SEC	1050101748301034	V0050050044
B	FOLLOW/R OUT	2383800272601024	V0002010045
B	PROC PROV1 (A)	347470007240111	H0018180046
B	PROC PROV1 (B)	335350007090111	H0003030047
B	PROC PROV1 (C)	335350011240111	H0014140048
B	PROC PROV1 (D)	323230007240111	H0018180049
B	PROC PROV1 (E)	311110019230111	H0005050050
B	PROC PROV2 (A)	346460007240111	H0018180051

B	PROC PROV2	(B)	334340007090111	H0003030052
B	PROC PROV2	(C)	334340011240111	H0014140053
B	PROC PROV2	(D)	322220007240111	H0018180054
B	PROC PROV2	(E)	310100019230111	H0005050055
BDX1	COL1	(A)	247470031330109	V0003000056
BDX1	COL1	(B)	247470035410109	V0007000057
BDX1	COL1	(C)	247470044570109	V0014000058
BDX1	COL1	(D)	247470059630109	V0005000059
BDX1	COL1	(E)	247470066720109	V0007000060
BDX1	COL1	(F)	247470075800109	V0006000061
BDX1	COL1	(G)	2474700828301024	V0002020062
BDX1	COL2	(A)	235350029300109	V0002000063
BDX1	COL2	(B)	235350033380109	V0006000064
BDX1	COL2	(C)	235350041490109	V0009000065
BDX1	COL2	(D)	235350052520009	V0001000066
BDX1	COL2	(E)	235350054560109	V0003000067
BDX1	COL2	(F)	235350059660109	V0008000068
BDX1	COL2	(G)	2353500698301024	V0015020069
BDX1	COL3	(A)	223230028290109	V0002000070
BDX1	COL3	(B)	223230032430109	V0012000071
BDX1	COL3	(C)	223230046550109	V0010000072
BDX1	COL3	(D)	223230058600109	V0003000073
BDX1	COL3	(E)	223230062680109	V0007000074
BDX1	COL3	(F)	223230071750109	V0005000075
BDX1	COL3	(G)	2232300778301024	V0007020076
BDX1	COL4	(A)	211110028300109	V0003000077
BDX1	COL4	(B)	211110033370109	V0005000078
BDX1	COL4	(C)	211110040430109	V0004000079
BDX1	COL4	(D)	211110046480109	V0003000080
BDX1	COL4	(E)	211110051650109	V0015000081
BDX1	COL4	(F)	2111100676801024	V0002020082
B	DX2	(A)	346460031330111	H0003030083
B	DX2	(B)	346460035410111	H0007070084
B	DX2	(C)	346460044570111	H0014140085
B	DX2	(D)	346460059630111	H0005050086
B	DX2	(E)	346460066720111	H0007070087
B	DX2	(F)	346460075800111	H0006060088
B	DX2	(G)	346460082830111	H0002020089
B	DX2	(H)	334340029300111	H0002020090
B	DX2	(I)	334340033380111	H0006060091
B	DX2	(J)	334340041490111	H0009090092
B	DX2	(K)	334340052520011	H0001010093
B	DX2	(L)	334340054560111	H0003030094
B	DX2	(M)	334340059660111	H0008080095
B	DX2	(N)	334340069830111	H0015150096
B	DX2	(O)	322220028290111	H0002020097
B	DX2	(P)	322220032430111	H0012120098
B	DX2	(Q)	322220046550111	H0010100099
B	DX2	(R)	322220058600111	H0003030100
B	DX2	(S)	322220062680111	H0007070101
B	DX2	(T)	322220071750111	H0005050102
B	DX2	(U)	322220077830111	H0007070103
B	DX2	(V)	310100028300111	H0003030104
B	DX2	(W)	310100033370111	H0005050105
B	DX2	(X)	310100040430111	H0004040106

B DX2 (Y) 310100046480111 H0003030107
B DX2 (Z) 310100051650111 H0015150108
//EOF0627

520 INFECTIOUS DISEASES

SCAN520.FIL

H03830247NY0051INFECTIOUS DISEASES *

H010245 *

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	223230017170011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800272601024V0002010044	
B	PROC PROV1 (A)	347470008100111	H0003030045
B	PROC PROV1 (B)	347470012130111	H0002020046
B	PROC PROV1 (C)	335350008120111	H0005050047
B	PROC PROV1 (D)	323230008140111	H0007070048
B	PROC PROV2 (A)	346460008100111	H0003030049
B	PROC PROV2 (B)	346460012130111	H0002020050
B	PROC PROV2 (C)	334340008120111	H0005050051

B	PROC PROV2 (D)	322220008140111	H0007070052
BDX1	COL1 (A)	247470031340109	V0004000053
BDX1	COL1 (B)	247470036370109	V0002000054
BDX1	COL1 (C)	247470039460109	V0008000055
BDX1	COL1 (D)	247470048540109	V0007000056
BDX1	COL1 (E)	2474700566401024	V0009020057
BDX1	COL2 (A)	235350029310109	V0003000058
BDX1	COL2 (B)	235350033360109	V0004000059
BDX1	COL2 (C)	235350038410109	V0004000060
BDX1	COL2 (D)	235350043470109	V0005000061
BDX1	COL2 (E)	2353500495601024	V0008020062
BDX1	COL3 (A)	223230029310109	V0003000063
BDX1	COL3 (B)	223230033370109	V0005000064
BDX1	COL3 (C)	223230039440109	V0006000065
BDX1	COL3 (D)	223230046460009	V0001000066
BDX1	COL3 (E)	223230048500109	V0003000067
BDX1	COL3 (F)	2232300525200024	V0001020068
B	DX2 (A)	346460031340111	H0004040069
B	DX2 (B)	346460036370111	H0002020070
B	DX2 (C)	346460039460111	H0008080071
B	DX2 (D)	346460048540111	H0007070072
B	DX2 (E)	346460056640111	H0009090073
B	DX2 (F)	334340029310111	H0003030074
B	DX2 (G)	334340033360111	H0004040075
B	DX2 (H)	334340038410111	H0004040076
B	DX2 (I)	334340043470111	H0005050077
B	DX2 (J)	334340049560111	H0008080078
B	DX2 (K)	322220029310111	H0003030079
B	DX2 (L)	322220033370111	H0005050080
B	DX2 (M)	322220039440111	H0006060081
B	DX2 (N)	322220046460011	H0001010082
B	DX2 (O)	322220048500111	H0003030083

//EOF0451

530 INTERNAL MEDICINE

SCAN530.FIL

H03830247NY0051INTERNAL MEDICINE

*

H010239

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA	2171205818100024H0002010005	
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY) 1	1454500555201014V0004010007	
B	DATE (DY) 2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	247470021210011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800272601024V0002010044	
B	PROC PROV1	347470007190111	H0013130045
B	PROC PROV2	346460007190111	H0013130046
B	DX1 COL1 (A)	247470031360109	V0006000047
B	DX1 COL1 (B)	247470039490109	V0011000048
B	DX1 COL1 (C)	247470052540109	V0003000049
E	DX1 COL1 (D)	2474700576101024V0005020050	
B	DX1 COL2 (A)	235350029370109	V0009000051

B	DX1 COL2 (B)	2353500405701024V0018020052
B	DX1 COL3 (A)	223230029350109 V0007000053
B	DX1 COL3 (B)	223230038410109 V0004000054
B	DX1 COL3 (C)	223230044450109 V0002000055
B	DX1 COL3 (D)	2232300485201024V0005020056
B	DX1 COL4 (A)	211110029290009 V0001000057
B	DX1 COL4 (B)	211110031310009 V0001000058
B	DX1 COL4 (C)	211110033340109 V0002000059
B	DX1 COL4 (D)	211110037390109 V0003000060
B	DX1 COL4 (E)	211110042420009 V0001000061
B	DX1 COL4 (F)	2111100444501024V0002020062
B	DX2 (A)	346460031360111 H0006060063
B	DX2 (B)	346460039490111 H0011110064
B	DX2 (C)	346460052540111 H0003030065
B	DX2 (D)	346460057610111 H0005050066
B	DX2 (E)	334340029370111 H0009090067
B	DX2 (F)	334340040570111 H0018180068
B	DX2 (G)	322220029350111 H0007070069
B	DX2 (H)	322220038410111 H0004040070
B	DX2 (I)	322220044450111 H0002020071
B	DX2 (J)	322220048520111 H0005050072
B	DX2 (K)	310100029290011 H0001010073
B	DX2 (L)	310100031310011 H0001010074
B	DX2 (M)	310100033340111 H0002020075
B	DX2 (N)	310100037390111 H0003030076
B	DX2 (O)	310100042420011 H0001010077

//EOF0449

H03830247NY0051NEPHROLOGY/DIALYSIS

*

H010217

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA	2171700818100024H0001010005	
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	247470027270011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800393801024V0002010044	
B	PROC PROV1 (A)	347470008100111	H0003030045
B	PROC PROV1 (B)	347470012150111	H0004040046
B	PROC PROV1 (C)	347470017190111	H0003030047
B	PROC PROV1 (D)	347470021250111	H0005050048
B	PROC PROV1 (E)	335350009090011	H0001010049
B	PROC PROV1 (F)	335350011130111	H0003030050
B	PROC PROV1 (G)	335350015160111	H0002020051

B	PROC	PROV1	(H)	335350018210111	H0004040052
B	PROC	PROV1	(I)	335350024250111	H0002020053
B	PROC	PROV1	(J)	335350027280111	H0002020054
B	PROC	PROV1	(K)	335350030310111	H0002020055
B	PROC	PROV1	(L)	335350033340111	H0002020056
B	PROC	PROV1	(M)	323230009190111	H0011110057
B	PROC	PROV1	(N)	323230021210011	H0001010058
B	PROC	PROV1	(O)	323230023240111	H0002020059
B	PROC	PROV1	(P)	323230026320111	H0007070060
B	PROC	PROV1	(Q)	311110021230111	H0003030061
B	PROC	PROV1	(R)	311110025280111	H0004040062
B	PROC	PROV1	(S)	311110030300011	H0001010063
B	PROC	PROV1	(T)	311110032330111	H0002020064
B	PROC	PROV2	(A)	346460008100111	H0003030065
B	PROC	PROV2	(B)	346460012150111	H0004040066
B	PROC	PROV2	(C)	346460017190111	H0003030067
B	PROC	PROV2	(D)	346460021250111	H0005050068
B	PROC	PROV2	(E)	334340009090011	H0001010069
B	PROC	PROV2	(F)	334340011130111	H0003030070
B	PROC	PROV2	(G)	334340015160111	H0002020071
B	PROC	PROV2	(H)	334340018210111	H0004040072
B	PROC	PROV2	(I)	334340024250111	H0002020073
B	PROC	PROV2	(J)	334340027280111	H0002020074
B	PROC	PROV2	(K)	334340030310111	H0002020075
B	PROC	PROV2	(L)	334340033340111	H0002020076
B	PROC	PROV2	(M)	322220009190111	H0011110077
B	PROC	PROV2	(N)	322220021210011	H0001010078
B	PROC	PROV2	(O)	322220023240111	H0002020079
B	PROC	PROV2	(P)	322220026320111	H0007070080
B	PROC	PROV2	(Q)	310100021230111	H0003030081
B	PROC	PROV2	(R)	310100025280111	H0004040082
B	PROC	PROV2	(S)	310100030300011	H0001010083
B	PROC	PROV2	(T)	310100032330111	H0002020084
B	OTHER CODES	(A)		322220072830111	H0012120085
B	OTHER CODES	(B)		318180072830111	H0012120086
B	DX1	COL1	(A)	247470044470109	V0004000087
B	DX1	COL1	(B)	247470049540109	V0006000088
B	DX1	COL1	(C)	247470056580109	V0003000089
B	DX1	COL1	(D)	247470060630109	V0004000090
B	DX1	COL1	(E)	247470065700109	V0006000091
B	DX1	COL1	(F)	2474700727901024	V0008020092
B	DX1	COL2	(A)	235350041450109	V0005000093
B	DX1	COL2	(B)	235350047480109	V0002000094
B	DX1	COL2	(C)	235350050550109	V0006000095
B	DX1	COL2	(D)	235350057610109	V0005000096
B	DX1	COL2	(E)	235350063670109	V0005000097
B	DX1	COL2	(F)	235350069760109	V0008000098
B	DX1	COL2	(G)	2353500788301024	V0006020099
B	DX1	COL3	(A)	223230041440109	V0004000100
B	DX1	COL3	(B)	223230046470109	V0002000101
B	DX1	COL3	(C)	223230050540109	V0005000102
B	DX1	COL3	(D)	223230056610109	V0006000103
B	DX1	COL3	(E)	2232300636300024	V0001020104
B	DX1	COL4	(A)	211110041430109	V0003000105
B	DX1	COL4	(B)	211110045460109	V0002000106

B	DX1 COL4 (C)	211110048500109	V0003000107
B	DX1 COL4 (D)	211110052540109	V0003000108
B	DX1 COL4 (E)	211110056560009	V0001000109
B	DX1 COL4 (F)	211110058590109	V0002000110
B	DX1 COL4 (G)	211110063630009	V0001000111
B	DX1 COL4 (H)	211110065650009	V0001000112
B	DX1 COL4 (I)	2111100686800024	V0001020113
B	DX2 (A)	346460044470111	H0004040114
B	DX2 (B)	346460049540111	H0006060115
B	DX2 (C)	346460056580111	H0003030116
B	DX2 (D)	346460060630111	H0004040117
B	DX2 (E)	346460065700111	H0006060118
B	DX2 (F)	346460072790111	H0008080119
B	DX2 (G)	334340041450111	H0005050120
B	DX2 (H)	334340047480111	H0002020121
B	DX2 (I)	334340050550111	H0006060122
B	DX2 (J)	334340057610111	H0005050123
B	DX2 (K)	334340063670111	H0005050124
B	DX2 (L)	334340069760111	H0008080125
B	DX2 (M)	334340078830111	H0006060126
B	DX2 (N)	322220041440111	H0004040127
B	DX2 (O)	322220046470111	H0002020128
B	DX2 (P)	322220050540111	H0005050129
B	DX2 (Q)	322220056610111	H0006060130
B	DX2 (R)	322220063630011	H0001010131
B	DX2 (S)	310100041430111	H0003030132
B	DX2 (T)	310100045460111	H0002020133
B	DX2 (U)	310100048500111	H0003030134
B	DX2 (V)	310100052540111	H0003030135
B	DX2 (W)	310100056560011	H0001010136
B	DX2 (X)	310100058590111	H0002020137
B	DX2 (Y)	310100063630011	H0001010138
B	DX2 (Z)	310100065650011	H0001010139

//EOF0599

H03830247NY0051NEUROLOGY

*

H010224

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA	2171700818100024V0001010005	
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY) 1	1454500555201014V0004010007	
B	DATE (DY) 2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	223230012120011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800151401024V0002010044	
B	PROC PROV1 (A)	347470007120111	H0006060045
B	PROC PROV1 (B)	335350007120111	H0006060046
B	PROC PROV1 (C)	323230007110111	H0005050047
B	PROC PROV2 (A)	346460007120111	H0006060048
B	PROC PROV2 (B)	334340007120111	H0006060049
B	PROC PROV2 (C)	322220007110111	H0005050050
B	DX1 COL1 (A)	247470019460109	V0028000051

B	DX1 COL1 (B)	247470049590109	V0011000052
B	DX1 COL1 (C)	247470062710109	V0010000053
B	DX1 COL1 (D)	2474700748301024	V0010020054
B	DX1 COL2 (A)	235350016240109	V0009000055
B	DX1 COL2 (B)	235350027500109	V0024000056
B	DX1 COL2 (C)	235350053590109	V0007000057
B	DX1 COL2 (D)	2353500628301024	V0022020058
B	DX1 COL3 (A)	223230017320109	V0016000059
B	DX1 COL3 (B)	223230034430109	V0010000060
B	DX1 COL3 (C)	223230046580109	V0013000061
B	DX1 COL3 (D)	2232300617101024	V0011020062
B	DX1 COL4 (A)	211110021310109	V0011000063
B	DX1 COL4 (B)	211110034480109	V0015000064
B	DX1 COL4 (C)	211110050600109	V0011000065
B	DX1 COL4 (D)	211110063640109	V0002000066
B	DX1 COL4 (E)	211110066660009	V0001000067
B	DX1 COL4 (F)	2111100686800024	V0001020068
B	DX2 (A)	346460019460111	H0028280069
B	DX2 (B)	346460049590111	H0011110070
B	DX2 (C)	346460062710111	H0010100071
B	DX2 (D)	346460074830111	H0010100072
B	DX2 (E)	334340016240111	H0009090073
B	DX2 (F)	334340027500111	H0024240074
B	DX2 (G)	334340053590111	H0007070075
B	DX2 (H)	334340062830111	H0022220076
B	DX2 (I)	322220017320111	H0016160077
B	DX2 (J)	322220034430111	H0010100078
B	DX2 (K)	322220046580111	H0013130079
B	DX2 (L)	322220061710111	H0011110080
B	DX2 (M)	310100021310111	H0011110081
B	DX2 (N)	310100034480111	H0015150082
B	DX2 (O)	310100050600111	H0011110083
B	DX2 (P)	310100063640111	H0002020084
B	DX2 (Q)	310100066660011	H0001010085

//EOF0590

560 NEUROSURGERY

SCAN560.FIL

H03830247NY0051NEUROSURGERY

*

H010236

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
	BIF NOT CLINIC (1)	217170063610109	V0003000023
	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	223230014140011	V0001010039
	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800181701024V0002010044	
B	PROC PROV1 (A)	347470008130111	H0006060045
B	PROC PROV1 (B)	347470015150011	H0001010046
B	PROC PROV1 (C)	335350007120111	H0006060047
B	PROC PROV1 (D)	335350014150111	H0002020048
B	PROC PROV1 (E)	323230007070011	H0001010049
B	PROC PROV1 (F)	323230009120111	H0004040050
B	PROC PROV2 (A)	346460008130111	H0006060051

B	PROC	PROV2	(B)	346460015150011	H0001010052
B	PROC	PROV2	(C)	334340007120111	H0006060053
B	PROC	PROV2	(D)	334340014150111	H0002020054
B	PROC	PROV2	(E)	322220007070011	H0001010055
B	PROC	PROV2	(F)	322220009120111	H0004040056
B	OTHER	(M)	CODES	311110040680111	H0029290057
B	DX1	COL1	(A)	247470022620109	V0041000058
B	DX1	COL1	(B)	247470064790109	V0016000059
B	DX1	COL1	(C)	2474700818301024	V0003020060
B	DX1	COL2	(A)	235350017260109	V0010000061
B	DX1	COL2	(B)	235350028490109	V0022000062
B	DX1	COL2	(C)	235350051540109	V0004000063
B	DX1	COL2	(D)	235350056630109	V0008000064
B	DX1	COL2	(E)	235350065710109	V0007000065
B	DX1	COL2	(F)	2353500738301024	V0011020066
B	DX1	COL3	(A)	223230017230109	V0007000067
B	DX1	COL3	(B)	223230025320109	V0008000068
B	DX1	COL3	(C)	223230034490109	V0016000069
B	DX1	COL3	(D)	223230051630109	V0013000070
B	DX1	COL3	(E)	223230065700109	V0006000071
B	DX1	COL3	(F)	223230072780109	V0007000072
B	DX1	COL3	(G)	2232300808301024	V0004020073
B	DX1	COL4	(A)	211110017190109	V0003000074
B	DX1	COL4	(B)	211110021260109	V0006000075
B	DX1	COL4	(C)	211110028310109	V0004000076
B	DX1	COL4	(D)	211110033340109	V0002000077
B	DX1	COL4	(E)	2111100363701024	V0002020078
B	DX2	(A)		346460022620111	H0041410079
B	DX2	(B)		346460064790111	H0016160080
B	DX2	(C)		346460081830111	H0003030081
B	DX2	(D)		334340017260111	H0010100082
B	DX2	(E)		334340028490111	H0022220083
B	DX2	(F)		334340051540111	H0004040084
B	DX2	(G)		334340056630111	H0008080085
B	DX2	(H)		334340065710111	H0007070086
B	DX2	(I)		334340073830111	H0011110087
B	DX2	(J)		322220017230111	H0007070088
B	DX2	(K)		322220025320111	H0008080089
B	DX2	(L)		322220034490111	H0016160090
B	DX2	(M)		322220051630111	H0013130091
B	DX2	(N)		322220065700111	H0006060092
B	DX2	(O)		322220072780111	H0007070093
B	DX2	(P)		322220080830111	H0004040094
B	DX2	(Q)		310100017190111	H0003030095
B	DX2	(R)		310100021260111	H0006060096
B	DX2	(S)		310100028310111	H0004040097
B	DX2	(T)		310100033340111	H0002020098

//EOF0612

580 NUTRITION

SCAN580.FIL

H02830247NY0051NUTRITION

*

H0346

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
F	JOB RELATED	801	010028
F	MIL ONLY DUTY	801	010029
F	MIL ONLY QTRS	801	010030
F	MIL ONLY PROF	801	010031
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
F	ADMITTED	801	010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800292801024V0002010044	
B	PROC PROV1 (A)	347470008120111	H0005050045
B	PROC PROV1 (B)	347470014160111	H0003030046
B	PROC PROV1 (C)	335350008110111	H0004040047
B	PROC PROV1 (D)	335350014160111	H0003030048
B	PROC PROV1 (E)	335350018220111	H0005050049
B	PROC PROV1 (F)	335350024240011	H0001010050
B	PROC PROV1 (G)	323230008090111	H0002020051

B	PROC	PROV1	(H)	323230011240111	H0014140052
B	PROC	PROV1	(I)	311110019190011	H0001010053
B	PROC	PROV2	(A)	346460008120111	H0005050054
B	PROC	PROV2	(B)	346460014160111	H0003030055
B	PROC	PROV2	(C)	334340008110111	H0004040056
B	PROC	PROV2	(D)	334340014160111	H0003030057
B	PROC	PROV2	(E)	334340018220111	H0005050058
B	PROC	PROV2	(F)	334340024240011	H0001010059
B	PROC	PROV2	(G)	322220008090111	H0002020060
B	PROC	PROV2	(H)	322220011240111	H0014140061
B	PROC	PROV2	(I)	310100019190011	H0001010062
BDX1	COL1	(A)		247470033400109	V0008000063
BDX1	COL1	(B)		247470043460109	V0004000064
BDX1	COL1	(C)		247470048540109	V0007000065
BDX1	COL1	(D)		2474700577901024	V0023020066
BDX1	COL2	(A)		235350031340109	V0004000067
BDX1	COL2	(B)		235350037420109	V0006000068
BDX1	COL2	(C)		235350045480109	V0004000069
BDX1	COL2	(D)		235350050520109	V0003000070
BDX1	COL2	(E)		235350053630109	V0011000071
BDX1	COL2	(F)		2353500667801024	V0013020072
BDX1	COL3	(A)		223230031360109	V0006000073
BDX1	COL3	(B)		223230039450109	V0007000074
BDX1	COL3	(C)		223230048640109	V0017000075
BDX1	COL3	(D)		2232300666701024	V0002020076
B	DX2	(A)		346460033400111	H0008080077
B	DX2	(B)		346460043460111	H0004040078
B	DX2	(C)		346460048540111	H0007070079
B	DX2	(D)		346460057790111	H0023230080
B	DX2	(E)		334340031340111	H0004040081
B	DX2	(F)		334340037420111	H0006060082
B	DX2	(G)		334340045480111	H0004040083
B	DX2	(H)		334340050520111	H0003030084
B	DX2	(I)		334340056630111	H0008080085
B	DX2	(J)		334340066780111	H0013130086
B	DX2	(K)		322220031360111	H0006060087
B	DX2	(L)		322220039450111	H0007070088
B	DX2	(M)		322220048640111	H0017170089

//EOF0529

590 OB

SCAN590.FIL

H03830247NY00510B

*

H010233

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
B	OTHER UCA1	217170081800109	V0002000005
B	OTHER UCA2	2121200818100024	H0001010006
B	DATE (MO)	2434300574601014	V0012020007
B	DATE (DY)1	1454500555201014	V0004010008
B	DATE (DY)2	1444400554601014	V0010010009
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014	V0013010011
B	PROV1 CODE	1423901332401014	V0040040012
B	SSN	1332501554601014	V0090090013
B	FMP	1373601554601014	V0020020014
B	VISIT COUNT	2141400544601024	V0009010015
B	PROV1 TIME	2373700351401014	V0022020016
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024	V0013010018
B	PROV2 CODE	1272401332401024	V0040040019
B	PROV2 TIME	2313100351401024	V0022020020
	BREASON FOR #2 (1)	230300011090109	V0003000021
	BREASON FOR #2 (2)	2272501080800024	H0003010022
B	IF NOT SCHED	2171700717001024	V0002010023
B	BIF NOT CLINIC (1)	217170063610109	V0003000024
B	BIF NOT CLINIC (2)	2141201606000024	H0003010025
B	REF CODE PFX	2040400706002024	V0006010026
B	REF CODE 1	0030300716001024	V0012010027
B	REF CODE 2	0020101714501024	V0054020028
B	JOB RELATED	217170032320011	V0001010029
B	MIL ONLY DUTY	2202000241410024	V0002010030
B	MIL ONLY QTRS	2191900222001024	V0003010031
B	MIL ONLY PROF	2191900181501024	V0004010032
F	FILLER	805	050033
B	SPEC PRE CLIN	306060022140111	H0009090034
B	NOT AVAILABLE	320200009070111	H0003030035
	BPROV2 ADDL1 (YES)	208080006060011	V0001010036
B	ADDL PROC1	1110701071601024	V0050050037
	BPROV2 ADDL2 (YES)	202020006060011	V0001010038
B	ADDL PROC2	1050101071601024	V0050050039
B	ADMITTED	223230023230011	V0001010040
	BUNL REAS PRIM PFX	2121200747602024	V0002010041
B	UNL REAS PRIM	1110701748301034	V0050050042
B	UNL REAS SEC PFX	2060600747602024	V0002010043
B	UNL REAS SEC	1050101748301034	V0050050044
B	FOLLOW/R OUT	2383800272601024	V0002010045
B	PROC PROV1 (A)	347470007080111	H0002020046
B	PROC PROV1 (B)	347470010220111	H0013130047
B	PROC PROV1 (C)	335350007140111	H0008080048
B	PROC PROV1 (D)	335350016220111	H0007070049
B	PROC PROV1 (E)	323230007200111	H0014140050
B	PROC PROV2 (A)	346460007080111	H0002020051

B	PROC PROV2	(B)	346460010220111	H0013130052
B	PROC PROV2	(C)	334340007140111	H0008080053
B	PROC PROV2	(D)	334340016220111	H0007070054
B	PROC PROV2	(E)	322220007200111	H0014140055
BDX1	COL1	(A)	247470031350109	V0005000056
BDX1	COL1	(B)	247470038440109	V0007000057
BDX1	COL1	(C)	247470047490109	V0003000058
BDX1	COL1	(D)	247470052540109	V0003000059
BDX1	COL1	(E)	247470057600109	V0004000060
BDX1	COL1	(F)	2474700636501024	V0003020061
BDX1	COL2	(A)	235350029310109	V0003000062
BDX1	COL2	(B)	235350034370109	V0004000063
BDX1	COL2	(C)	235350040430109	V0004000064
BDX1	COL2	(D)	235350046490109	V0004000065
BDX1	COL2	(E)	235350052550109	V0004000066
BDX1	COL2	(F)	2353500586201024	V0005020067
BDX1	COL3	(G)	2232300296201024	V0034020068
BDX1	COL4	(A)	211110029340109	V0006000069
BDX1	COL4	(B)	211110036390109	V0004000070
BDX1	COL4	(C)	2111100424301024	V0002020071
B	DX2	(A)	346460031350111	H0005050072
B	DX2	(B)	346460038440111	H0007070073
B	DX2	(C)	346460047490111	H0003030074
B	DX2	(D)	346460052540111	H0003030075
B	DX2	(E)	346460057600111	H0004040076
B	DX2	(F)	346460063650111	H0003030077
B	DX2	(G)	334340029310111	H0003030078
B	DX2	(H)	334340034370111	H0004040079
B	DX2	(I)	334340040430111	H0004040080
B	DX2	(J)	334340046490111	H0004040081
B	DX2	(K)	334340052550111	H0004040082
B	DX2	(L)	334340058620111	H0005050083
B	DX2	(M)	322220029620111	H0034340084
B	DX2	(N)	310100029340111	H0006060085
B	DX2	(O)	310100036390111	H0004040086

//EOF0526

600 ONCOLOGY

SCAN600.FIL

H02830247NY0051ONCOLOGY

*

H0344

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA 1	217170081800109	V0002000005
B	OTHER UCA 2	2121200818100024V0001010006	
B	DATE (MO)	2434300574601014V0012020007	
B	DATE (DY) 1	1454500555201014V0004010008	
B	DATE (DY) 2	1444400554601014V0010010009	
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014V0013010011	
B	PROV1 CODE	1423901332401014V0040040012	
B	SSN	1332501554601014V0090090013	
B	FMP	1373601554601014V0020020014	
B	VISIT COUNT	2141400544601024V0009010015	
B	PROV1 TIME	2373700351401014V0022020016	
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024V0013010018	
B	PROV2 CODE	1272401332401024V0040040019	
B	PROV2 TIME	2313100351401024V0022020020	
B	BREASON FOR #2 (1)	230300011090109	V0003000021
B	BREASON FOR #2 (2)	2272501080800024H0003010022	
B	IF NOT SCHED	2171700717001024V0002010023	
B	BIF NOT CLINIC (1)	217170063610109	V0003000024
B	BIF NOT CLINIC (2)	2141201606000024H0003010025	
B	REF CODE PFX	2040400706002024V0006010026	
B	REF CODE 1	0030300716001024V0012010027	
B	REF CODE 2	0020101714501024V0054020028	
F	JOB RELATED	801	010029
F	MIL ONLY DUTY	801	010030
F	MIL ONLY QTRS	801	010031
F	MIL ONLY PROF	801	010032
F	FILLER	805	050033
B	SPEC PRE CLIN	306060022140111	H0009090034
B	NOT AVAILABLE	320200009070111	H0003030035
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010036
B	ADDL PROC1	1110701071601024V0050050037	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010038
B	ADDL PROC2	1050101071601024V0050050039	
B	ADMITTED	223230019190011	V0001010040
B	BUNL REAS PRIM PFX	2121200747602024V0002010041	
B	UNL REAS PRIM	1110701748301034V0050050042	
B	UNL REAS SEC PFX	2060600747602024V0002010043	
B	UNL REAS SEC	1050101748301034V0050050044	
B	FOLLOW/R OUT	2383800272601024V0002010045	
B	PROC PROV1 (A)	347470007140111	H0008080046
B	PROC PROV1 (B)	335350007080111	H0002020047
B	PROC PROV1 (C)	335350010160111	H0007070048
B	PROC PROV1 (D)	323230008080011	H0001010049
B	PROC PROV1 (E)	323230010100011	H0001010050
B	PROC PROV1 (F)	323230012150111	H0004040051

B	PROC	PROV1	(G)	323230017170011	H0001010052
B	PROC	PROV2	(A)	346460007140111	H0008080053
B	PROC	PROV2	(B)	334340007080111	H0002020054
B	PROC	PROV2	(C)	334340010160111	H0007070055
B	PROC	PROV2	(D)	322220008080011	H0001010056
B	PROC	PROV2	(E)	322220010100011	H0001010057
B	PROC	PRCV2	(F)	322220012150111	H0004040058
B	PROC	PROV2	(G)	322220017170011	H0001010059
B	DX1	COL1	(A)	247470031440109	V0014000060
B	DX1	COL1	(B)	247470047550109	V0009000061
B	DX1	COL1	(C)	247470058600109	V0003000062
B	DX1	COL1	(D)	2474700637601024	V0014020063
B	DX1	COL2	(A)	235350029310109	V0003000064
B	DX1	COL2	(B)	235350034470109	V0014000065
B	DX1	COL2	(C)	235350049510109	V0003000066
B	DX1	COL2	(D)	235350053570109	V0005000067
B	DX1	COL2	(E)	235350059590009	V0001000068
B	DX1	COL2	(F)	235350061610009	V0001000069
B	DX1	COL2	(G)	235350063640109	V0002000070
B	DX1	COL2	(H)	2353500667101024	V0006020071
B	DX1	COL3	(A)	223230029350109	V0007000072
B	DX1	COL3	(B)	223230037420109	V0006000073
B	DX1	COL3	(C)	223230045490109	V0005000074
B	DX1	COL3	(D)	223230052550109	V0004000075
B	DX1	COL3	(E)	223230058650109	V0008000076
B	DX1	COL3	(F)	223230068710109	V0004000077
B	DX1	COL3	(G)	223230073740109	V0002000078
B	DX1	COL3	(H)	2232300767801024	V0003020079
B	DX1	COL4	(A)	211110029300109	V0002000080
B	DX1	COL4	(B)	211110032320009	V0001000081
B	DX1	COL4	(C)	211110035600109	V0026000082
B	DX1	COL4	(D)	211110063660109	V0004000083
B	DX1	COL4	(E)	2111100686800024	V0001020084
B	DX2	(A)		346460031440111	H0014140085
B	DX2	(B)		346460047550111	H0009090086
B	DX2	(C)		346460058600111	H0003030087
B	DX2	(D)		346460063760111	H0014140088
B	DX2	(E)		334340029310111	H0003030089
B	DX2	(F)		334340034470111	H0014140090
B	DX2	(G)		334340049510111	H0003030091
B	DX2	(H)		334340053570111	H0005050092
B	DX2	(I)		334340059590011	H0001010093
B	DX2	(J)		334340061610011	H0001010094
B	DX2	(K)		334340063640111	H0002020095
B	DX2	(L)		334340066710111	H0006060096
B	DX2	(M)		322220029350111	H0007070097
B	DX2	(N)		322220037420111	H0006060098
B	DX2	(O)		322220045490111	H0005050099
B	DX2	(P)		322220052550111	H0004040100
B	DX2	(Q)		322220058650111	H0008080101
B	DX2	(R)		322220068710111	H0004040102
B	DX2	(S)		322220073740111	H0002020103
B	DX2	(T)		322220076780111	H0003030104
B	DX2	(U)		310100029300111	H0002020105
B	DX2	(V)		310100032320011	H0001010106

B DX2 (W)
B DX2 (X)
//EOF0538

310100035600111 H0026260107
310100063660111 H0004040108

H03830247NY0051OPHTHAMOLOGY

*

H010219

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	223230023230011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800282701024V0002010044	
B	PROC PROV1 (A)	347470008080011	H0001010045
B	PROC PROV1 (B)	347470011120111	H0002020046
B	PROC PROV1 (C)	347470015240111	H0010100047
B	PROC PROV1 (D)	335350008100111	H0003030048
B	PROC PROV1 (E)	335350013240111	H0012120049
B	PROC PROV1 (F)	323230007180111	H0012120050
B	PROC PROV1 (G)	323230020200011	H0001010051

B	PROC	PROV2	(A)	346460008080011	H0001010052
B	PROC	PROV2	(B)	346460011120111	H0002020053
B	PROC	PROV2	(C)	346460015240111	H0010100054
B	PROC	PROV2	(D)	334340008100111	H0003030055
B	PROC	PROV2	(E)	334340013240111	H0012120056
B	PROC	PROV2	(F)	322220007180111	H0012120057
B	PROC	PROV2	(G)	322220020200011	H0001010058
B	DX1	COL1	(A)	247470033430109	V0011000059
B	DX1	COL1	(B)	2474700466401024	V0019020060
B	DX1	COL2	(A)	235350030430109	V0014000061
B	DX1	COL2	(B)	235350046510109	V0006000062
B	DX1	COL2	(C)	2353500546101024	V0008020063
B	DX1	COL3	(A)	223230030450109	V0016000064
B	DX1	COL3	(B)	223230048570109	V0010000065
B	DX1	COL3	(C)	2232300596401024	V0006020066
B	DX1	COL4	(A)	211110030310109	V0002000067
B	DX1	COL4	(B)	211110034360109	V0003000068
B	DX1	COL4	(C)	211110039490109	V0011000069
B	DX1	COL4	(D)	2111100515201024	V0002020070
B	DX2	(A)		346460033430111	H0011110071
B	DX2	(B)		346460046640111	H0019190072
B	DX2	(C)		334340030430111	H0014140073
B	DX2	(D)		334340046510111	H0006060074
B	DX2	(E)		334340054610111	H0008080075
B	DX2	(F)		322220030450111	H0016160076
B	DX2	(G)		322220048570111	H0010100077
B	DX2	(H)		322220059640111	H0006060078
B	DX2	(I)		310100030310111	H0002020079
B	DX2	(J)		310100034360111	H0003030080
B	DX2	(K)		310100039490111	H0011110081

//EOF0533

620 OPTOMETRY

SCAN620.FIL

H03830247NY0051OPTOMETRY

*

H010246

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE	0080601714501024	V0081030003
B	OTHER UCA	2171205818100024	H0002010004
B	DATE (MO)	2434300574601014	V0012020005
B	DATE (DY)	1454401554601014	V0020020006
B	PROV1 PFX1	044440033200109	V0014000007
B	PROV1 PFX2	0434300322001014	V0013010008
B	PROV1 CODE	1423901332401014	V0040040009
B	SSN	1332501554601014	V0090090010
B	FMP	1373601554601014	V0020020011
B	VISIT COUNT	2141400544601024	V0009010012
B	PROV1 TIME	2373700351401014	V0022020013
B	PROV2 PFX1	029290033200109	V0014000014
B	PROV2 PFX2	0282800322001024	V0013010015
B	PROV2 CODE	1272401332401024	V0040040016
B	PROV2 TIME	2313100351401024	V0022020017
BREASON FOR #2 (1)		230300011090109	V0003000018
BREASON FOR #2 (2)		2272501080800024	H0003010019
B	IF NOT SCHED	2171700717001024	V0002010020
BIF NOT CLINIC (1)		217170063610109	V0003000021
BIF NOT CLINIC (2)		2141201606000024	H0003010022
B	REF CODE PFX	2040400706002024	V0006010023
B	REF CODE	0030101714501024	V0081030024
B	JOB RELATED	217170032320011	V0001010025
B	MIL ONLY DUTY	2202000241410024	V0002010026
B	MIL ONLY QTRS	2191900222001024	V0003010027
B	MIL ONLY PROF	2191900181501024	V0004010028
F	FILLER	805	050029
B	SPEC PRE CLIN	306060022140111	H0009090030
B	NOT AVAILABLE	320200009070111	H0003030031
BPROV2 ADDL1 (YES)		208080006060011	V0001010032
B	ADDL PROC1	1110701071601024	V0050050033
BPROV2 ADDL2 (YES)		202020006060011	V0001010034
B	ADDL PROC2	1050101071601024	V0050050035
F	ADMITTED	801	010036
BUNL REAS PRIM PFX		2121200747602024	V0002010037
B	UNL REAS PRIM	1110701748301034	V0050050038
B	UNL REAS SEC PFX	2060600747602024	V0002010039
B	UNL REAS SEC	1050101748301034	V0050050040
B	FOLLOW/R OUT	2383800403901024	V0002010041
B	PROC PROV1 (A)	347470008110111	H0004040042
B	PROC PROV1 (B)	347470014180111	H0005050043
B	PROC PROV1 (C)	347470021240111	H0004040044
B	PROC PROV1 (D)	347470027280111	H0002020045
B	PROC PROV1 (E)	347470031330111	H0003030046
B	PROC PROV1 (F)	335350008100111	H0003030047
B	PROC PROV1 (G)	335350013310111	H0019190048
B	PROC PROV1 (H)	323230008150111	H0008080049
B	PROC PROV1 (I)	323230018200111	H0003030050
B	PROC PROV2 (A)	346460008110111	H0004040051

B	PROC PROV2 (B)	346460014180111	H0005050052
B	PROC PROV2 (C)	346460021240111	H0004040053
B	PROC PROV2 (D)	346460027280111	H0002020054
B	PROC PROV2 (E)	346460031330111	H0003030055
B	PROC PROV2 (F)	334340008100111	H0003030056
B	PROC PROV2 (G)	334340013310111	H0019190057
B	PROC PROV2 (H)	322220008150111	H0008080058
B	PROC PROV2 (I)	322220018200111	H0003030059
B	OTHER CODES (A)	323230023290111	H0007070060
B	OTHER CODES (B)	311110018230111	H0006060061
B	OTHER CODES (C)	311110026330111	H0008080062
B	DX1 COL1 (A)	247470044530109	V0010000063
B	DX1 COL1 (B)	247470056640109	V0009000064
B	DX1 COL1 (C)	247470067760109	V0010000065
B	DX1 COL1 (D)	2474700798301024	V0005020066
B	DX1 COL2 (A)	235350042470109	V0006000067
B	DX1 COL2 (B)	235350051580109	V0008000068
B	DX1 COL2 (C)	2353500617601024	V0016020069
B	DX1 COL3 (A)	223230042450109	V0004000070
B	DX1 COL3 (B)	223230047560109	V0010000071
B	DX1 COL3 (C)	223230059630109	V0005000072
B	DX1 COL3 (D)	223230066710109	V0006000073
B	DX1 COL3 (E)	2232300737300024	V0001020074
B	DX1 COL4 (A)	211110042430109	V0002000075
B	DX1 COL4 (B)	211110045450009	V0001000076
B	DX1 COL4 (C)	211110048480009	V0001000077
B	DX1 COL4 (D)	211110050550109	V0006000078
B	DX1 COL4 (E)	211110057570009	V0001000079
B	DX1 COL4 (F)	211110059610109	V0003000080
B	DX1 COL4 (G)	2111100646501024	V0002010081
B	DX2 (A)	346460044530111	H0010100082
B	DX2 (B)	346460056640111	H0009090083
B	DX2 (C)	346460067760111	H0010100084
B	DX2 (D)	346460079830111	H0005050085
B	DX2 (E)	334340042470111	H0006060086
B	DX2 (F)	334340051580111	H0008080087
B	DX2 (G)	334340061760111	H0016160088
B	DX2 (H)	322220042450111	H0004040089
B	DX2 (I)	322220047560111	H0010100090
B	DX2 (J)	322220059630111	H0005050091
B	DX2 (K)	322220066710111	H0006060092
B	DX2 (L)	322220073730011	H0001010093
B	DX2 (M)	310100042430111	H0002020094
B	DX2 (N)	310100045450011	H0001010095
B	DX2 (O)	310100048480011	H0001010096
B	DX2 (P)	310100050550111	H0006060097
B	DX2 (Q)	310100057570011	H0001010098
B	DX2 (R)	310100059610111	H0003030099

//EOF0344

H02830247NY0051ORTHO APPLIANCE/CAST

*

H0342

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA	2171700818100024V0001010005	
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY) 1	1454500555201014V0004010007	
B	DATE (DY) 2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
B	BREASON FOR #2 (1)	230300011090109	V0003000020
B	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
F	JOB RELATED	801	010028
F	MIL ONLY DUTY	801	010029
F	MIL ONLY QTRS	801	010030
F	MIL ONLY PROF	801	010031
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
F	ADMITTED	801	010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
F	FOLLOW/R OUT	801	010044
B	PROC PROV1 (A)	347470009210111	H0013130045
B	PROC PROV1 (B)	347470024270111	H0004040046
B	PROC PROV1 (C)	347470030340111	H0005050047
B	PROC PROV1 (D)	347470037440111	H0008080048
B	PROC PROV1 (E)	347470047510111	H0005050049
B	PROC PROV1 (F)	347470053550111	H0003030050
B	PROC PROV1 (G)	335350008100111	H0003030051

B	PROC PROV2 (A)	346460009210111	H0013130052
B	PROC PROV2 (B)	346460024270111	H0004040053
B	PROC PROV2 (C)	346460030340111	H0005050054
B	PROC PROV2 (D)	346460037440111	H0008080055
B	PROC PROV2 (E)	346460047510111	H0005050056
B	PROC PROV2 (F)	346460053550111	H0003030057
B	PROC PROV2 (G)	334340008100111	H0003030058
B	OTHER CODES (A)	335350017210111	H0005050059
B	OTHER CODES (B)	335350026410111	H0016160060
B	OTHER CODES (C)	335350043560111	H0014140061
B	OTHER CODES (D)	335350059620111	H0004040062
B	OTHER CODES (E)	323230026340111	H0009090063
B	OTHER CODES (F)	323230037550111	H0019190064
B	OTHER CODES (G)	323230058620111	H0005050065
B	OTHER CODES (H)	311110026350111	H0010100066
B	OTHER CODES (I)	311110038490111	H0012120067
B	OTHER CODES (J)	311110052640111	H0013130068
B	DX1 COL1 (A)	247470070710109	V0002000069
B	DX1 COL1 (B)	2474700738201024	V0010020070
B	DX1 COL2	2353500708001024	V0011020071
B	DX2 (A)	346460070710111	H0002020072
B	DX2 (B)	346460073820111	H0010100073
B	DX2 (C)	334340070800111	H0011110074

//EOF0553

640 ORTHOPEDICS

SCAN640.FIL

H03830247NY0051ORTHOPEDICS

*

H010227

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
B	OTHER UCA 1	217170081810009	V0001000005
B	OTHER UCA 2	2121200818100024	V0001010006
B	DATE (MO)	2434300574601014	V0012020007
B	DATE (DY)1	1454500555201014	V0004010008
B	DATE (DY)2	1444400554601014	V0010010009
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014	V0013010011
B	PROV1 CODE	1423901332401014	V0040040012
B	SSN	1332501554601014	V0090090013
B	FMP	1373601554601014	V0020020014
B	VISIT COUNT	2141400544601024	V0009010015
B	PROV1 TIME	2373700351401014	V0022020016
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024	V0013010018
B	PROV2 CODE	1272401332401024	V0040040019
B	PROV2 TIME	2313100351401024	V0022020020
BREASON FOR #2 (1)		230300011090109	V0003000021
BREASON FOR #2 (2)		2272501080800024	H0003010022
B	IF NOT SCHED	2171700717001024	V0002010023
BIF NOT CLINIC (1)		217170063610109	V0003000024
BIF NOT CLINIC (2)		2141201606000024	H0003010025
B	REF CODE PFX	2040400706002024	V0006010026
B	REF CODE 1	0030300716001024	V0012010027
B	REF CODE 2	0020101714501024	V0054020028
B	JOB RELATED	217170032320011	V0001010029
B	MIL ONLY DUTY	2202000241410024	V0002010030
B	MIL ONLY QTRS	2191900222001024	V0003010031
B	MIL ONLY PROF	2191900181501024	V0004010032
F	FILLER	805	050033
B	SPEC PRE CLIN	306060022140111	H0009090034
B	NOT AVAILABLE	320200009070111	H0003030035
BPROV2 ADDL1 (YES)		208080006060011	V0001010036
B	ADDL PROC1	1110701071601024	V0050050037
BPROV2 ADDL2 (YES)		202020006060011	V0001010038
B	ADDL PROC2	1050101071601024	V0050050039
B	ADMITTED	223230021210011	V0001010040
BUNL REAS PRIM PFX		2121200747602024	V0002010041
B	UNL REAS PRIM	1110701748301034	V0050050042
B	UNL REAS SEC PFX	2060600747602024	V0002010043
B	UNL REAS SEC	1050101748301034	V0050050044
B	FOLLOW/R OUT	2383800272601024	V0002010045
B	PROC PROV1 (A)	347470008120111	H0005050046
B	PROC PROV1 (B)	335350008130111	H0006060047
B	PROC PROV1 (C)	335350016180111	H0003030048
B	PROC PROV1 (D)	323230008170111	H0010100049
B	PROC PROV1 (E)	323230019190011	H0001010050
B	PROC PROV1 (A)	346460008120111	H0005050051

B	PROC	PROV1	(B)	334340008130111	H0006060052
B	PROC	PROV1	(C)	334340016180111	H0003030053
B	PROC	PROV1	(D)	322220008170111	H0010100054
B	PROC	PROV1	(E)	322220019190011	H0001010055
B	DX1	COL1	(A)	247470031490109	V0019000056
B	DX1	COL1	(B)	247470052540109	V0003000057
B	DX1	COL1	(C)	2474700566801024	V0013020058
B	DX1	COL2	(A)	235350029440109	V0016000059
B	DX1	COL2	(B)	235350047490109	V0003000060
B	DX1	COL2	(C)	235350051560109	V0006000061
B	DX1	COL2	(D)	2353500597601024	V0018020062
B	DX1	COL3	(A)	223230029380109	V0010000063
B	DX1	COL3	(B)	223230040410109	V0002000064
B	DX1	COL3	(C)	223230044490109	V0006000065
B	DX1	COL3	(D)	2232300516501024	V0015020066
B	DX1	COL4	(A)	211110029450109	V0017000067
B	DX1	COL4	(B)	211110047510109	V0005000068
B	DX1	COL4	(C)	2111100535300024	V0001020069
B		DX2	(A)	346460031490111	H0019190070
B		DX2	(B)	346460052540111	H0003030071
B		DX2	(C)	346460056680111	H0013130072
B		DX2	(D)	334340029440111	H0016160073
B		DX2	(E)	334340047490111	H0003030074
B		DX2	(F)	334340051560111	H0006060075
B		DX2	(G)	334340059760111	H0018180076
B		DX2	(H)	322220029380111	H0010100077
B		DX2	(I)	322220040410111	H0002020078
B		DX2	(J)	322220044490111	H0006060079
B		DX2	(K)	322220051650111	H0015150080
B		DX2	(L)	310100029450111	H0017170081
B		DX2	(M)	310100047510111	H0005050082

//EOF0530

650 OCCUPATIONAL THERAPY

SCAN650.FIL

H03830247NY0051OCCUPATIONAL THERAPY

*

H010220

*

B	LITHO	739160104040011	H0024090001
F	UCA PFX	801	010002
F	UCA CODE 1	801	010003
F	UCA CODE 2	802	020004
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014	V0012020006
B	DATE (DY)1	1454500555201014	V0004010007
B	DATE (DY)2	1444400554601014	V0010010008
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014	V0013010010
B	PROV1 CODE	1423901332401014	V0040040011
B	SSN	1332501554601014	V0090090012
B	FMP	1373601554601014	V0020020013
B	VISIT COUNT	2141400544601024	V0009010014
B	PROV1 TIME	2373700351401014	V0022020015
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024	V0013010017
B	PROV2 CODE	1272401332401024	V0040040018
B	PROV2 TIME	2313100351401024	V0022020019
BREASON FOR #2 (1)		230300011090109	V0003000020
BREASON FOR #2 (2)		227250108080009	H0003000021
BREASON FOR #2 (CO)		2303000121200024	V0001010022
B	IF NOT SCHED	2171700717001024	V0002010023
BIF NOT CLINIC (1)		217170063610109	V0003000024
BIF NOT CLINIC (2)		2141201606000024	H0003010025
B	REF CODE PFX	2040400706002024	V0006010026
B	REF CODE 1	0030300716001024	V0012010027
B	REF CODE 2	0020101714501024	V0054020028
B	JOB RELATED	217170032320011	V0001010029
B	MIL ONLY DUTY	2202000241410024	V0002010030
B	MIL ONLY QTRS	2191900222001024	V0003010031
B	MIL ONLY PROF	2191900181501024	V0004010032
B	REF CODE SING	2090900704501024	V0026020033
F	FILLER	803	030034
B	SPEC PRE CLIN	306060022140111	H0009090035
B	NOT AVAILABLE	320200009070111	H0003030036
BPROV2 ADDL1 (YES)		208080006060011	V0001010037
B	ADDL PROC1	1110701071601024	V0050050038
BPROV2 ADDL2 (YES)		202020006060011	V0001010039
B	ADDL PROC2	1050101071601024	V0050050040
F	ADMITTED	801	010041
BUNL REAS PRIM PFX		2121200747602024	V0002010042
B	UNL REAS PRIM	1110701748301034	V0050050043
B	UNL REAS SEC PFX	2060600747602024	V0002010044
B	UNL REAS SEC	1050101748301034	V0050050045
B	FOLLOW/R OUT	2383800494801024	V0002010046
B	PROC PROV1 (A)	347470008090111	H0002020047
B	PROC PROV1 (B)	347470011150111	H0005050048
B	PROC PROV1 (C)	347470017260111	H0010100049
B	PROC PROV1 (D)	347470028330111	H0006060050
B	PROC PROV1 (E)	347470035420111	H0008080051

B	PROC	PROV1	(F)	335350008170111	H0010100052
B	PROC	PROV1	(G)	335350019250111	H0007070053
B	PROC	PROV1	(H)	335350027320111	H0006060054
B	PROC	PROV1	(I)	335350036420111	H0007070055
B	PROC	PROV1	(J)	323230008110111	H0004040056
B	PROC	PROV1	(K)	323230013140111	H0002020057
B	PROC	PROV1	(L)	323230016270111	H0012120058
B	PROC	PROV1	(M)	323230029370111	H0009090059
B	PROC	PROV1	(N)	323230039460111	H0008080060
B	PROC	PROV1	(O)	311110019230111	H0005050061
B	PROC	PROV1	(P)	311110025330111	H0009090062
B	PROC	PROV1	(Q)	311110036450111	H0010100063
B	PROC	PROV2	(A)	346460008090111	H0002020064
B	PROC	PROV2	(B)	346460011150111	H0005050065
B	PROC	PROV2	(C)	346460017260111	H0010100066
B	PROC	PROV2	(D)	346460028330111	H0006060067
B	PROC	PROV2	(E)	346460035420111	H0008080068
B	PROC	PROV2	(F)	334340008170111	H0010100069
B	PROC	PROV2	(G)	334340019250111	H0007070070
B	PROC	PROV2	(H)	334340027320111	H0006060071
B	PROC	PROV2	(I)	334340036420111	H0007070072
B	PROC	PROV2	(J)	322220008110111	H0004040073
B	PROC	PROV2	(K)	322220013140111	H0002020074
B	PROC	PROV2	(L)	322220016270111	H0012120075
B	PROC	PROV2	(M)	322220029370111	H0009090076
B	PROC	PROV2	(N)	322220039460111	H0008080077
B	PROC	PROV2	(O)	310100019230111	H0005050078
B	PROC	PROV2	(P)	310100025330111	H0009090079
B	PROC	PROV2	(Q)	310100036450111	H0010100080
B	DX1	COL1	(A)	247470053690109	V0017000081
B	DX1	COL1	(B)	247470072760109	V0005000082
B	DX1	COL1	(C)	2474700788301024	V0006020083
B	DX1	COL2	(A)	235350050540109	V0005000084
B	DX1	COL2	(B)	2353500578201024	V0026020085
B	DX1	COL3	(A)	223230051720109	V0022000086
B	DX1	COL3	(B)	223230074780109	V0005000087
B	DX1	COL3	(C)	2232300808301024	V0004020088
B	DX1	COL4	(A)	211110050510109	V0002000089
B	DX1	COL4	(B)	211110053570109	V0005000090
B	DX1	COL4	(C)	211110060650109	V0006000091
B	DX1	COL4	(D)	2111100676801024	V0002020092
B		DX2	(A)	346460053690111	H0017170093
B		DX2	(B)	346460072760111	H0005050094
B		DX2	(C)	346460078830111	H0006060095
B		DX2	(D)	334340050540111	H0005050096
B		DX2	(E)	334340057820111	H0026260097
B		DX2	(F)	322220051720111	H0022220098
B		DX2	(G)	322220074780111	H0005050099
B		DX2	(H)	322220080830111	H0004040100
B		DX2	(I)	310100050510111	H0002020101
B		DX2	(J)	310100053570111	H0005050102
B		DX2	(K)	310100060650111	H0006060103

//EOF0683 *

```

H03830247NY0051PAIN/PHYSICAL MEDICINE      *
H010232                                         *
B          LITHO          739160104040011 H0024080001
B          UCA PFX        2090900706202024V0005010002
B          UCA CODE 1     0080800716001024V0012010003
B          UCA CODE 2     0070601714501024V0054020004
B          OTHER UCA      2171700818100024V0001010005
B          DATE (MO)      2434300574601014V0012020006
B          DATE (DY)1     1454500555201014V0004010007
B          DATE (DY)2     1444400554601014V0010010008
B          PROV1 PFX1     044440033200109 V0014000009
B          PROV1 PFX2     0434300322001014V0013010010
B          PROV1 CODE     1423901332401014V0040040011
B          SSN            1332501554601014V0090090012
B          FMP            1373601554601014V0020020013
B          VISIT COUNT    2141400544601024V0009010014
B          PROV1 TIME     2373700351401014V0022020015
B          PROV2 PFX1     029290033200109 V0014000016
B          PROV2 PFX2     0282800322001024V0013010017
B          PROV2 CODE     1272401332401024V0040040018
B          PROV2 TIME     2313100351401024V0022020019
BREASON FOR #2 (1)       230300011090109 V0003000020
BREASON FOR #2 (2)       2272501080800024H0003010021
B          IF NOT SCHED    2171700717001024V0002010022
BIF NOT CLINIC (1)       217170063610109 V0003000023
BIF NOT CLINIC (2)       2141201606000024H0003010024
B          REF CODE PFX    2040400706002024V0006010025
B          REF CODE 1      0030300716001024V0012010026
B          REF CODE 2      0020101714501024V0054020027
B          JOB RELATED     217170032320011 V0001010028
B          MIL ONLY DUTY   2202000241410024V0002010029
B          MIL ONLY QTRS   2191900222001024V0003010030
B          MIL ONLY PROF   2191900181501024V0004010031
F          FILLER         805                      050032
B          SPEC PRE CLIN   306060022140111 H0009090033
B          NOT AVAILABLE   320200009070111 H0003030034
BPROV2 ADDL1 (YES)       208080006060011 V0001010035
B          ADDL PROC1      1110701071601024V0050050036
BPROV2 ADDL2 (YES)       202020006060011 V0001010037
B          ADDL PROC2      1050101071601024V0050050038
B          ADMITTED        235350024240011 V0001010039
BUNL REAS PRIM PFX       2121200747602024V0002010040
B          UNL REAS PRIM   1110701748301034V0050050041
B          UNL REAS SEC PFX 2060600747602024V0002010042
B          UNL REAS SEC    1050101748301034V0050050043
B          FOLLOW/R OUT     2383800343301024V0002010044
B          PROC PROV1 (A)   347470010140111 H0005050045
B          PROC PROV1 (B)   347470016170111 H0002020046
B          PROC PROV1 (C)   347470019210111 H0003030047
B          PROC PROV1 (D)   335350011140111 H0004040048
B          PROC PROV1 (E)   335350016200111 H0005050049
B          PROC PROV1 (F)   335350022220011 H0001010050
B          PROC PROV1 (G)   323230009210111 H0013130051

```

B	PROC	PROV1	(H)	323230023300111	H0008080052
B	PROC	PROV1	(I)	311110020220111	H0003030053
B	PROC	PROV1	(J)	311110024280111	H0005050054
B	PROC	PROV1	(K)	311110030300011	H0001010055
B	PROC	PROV2	(A)	346460010140111	H0005050056
B	PROC	PROV2	(B)	346460016170111	H0002020057
B	PROC	PROV2	(C)	346460019210111	H0003030058
B	PROC	PROV2	(D)	334340011140111	H0004040059
B	PROC	PROV2	(E)	334340016200111	H0005050060
B	PROC	PROV2	(F)	334340022220011	H0001010061
B	PROC	PROV2	(G)	322220009210111	H0013130062
B	PROC	PROV2	(H)	322220023300111	H0008080063
B	PROC	PROV2	(I)	310100020220111	H0003030064
B	PROC	PROV2	(J)	310100024280111	H0005050065
B	PROC	PROV2	(K)	310100030300011	H0001010066
B	DX1	COL1	(A)	247470038420109	V0005000067
B	DX1	COL1	(B)	247470044480109	V0005000068
B	DX1	COL1	(C)	247470050510109	V0002000069
B	DX1	COL1	(D)	247470053550109	V0003000070
B	DX1	COL1	(E)	247470057600109	V0004000071
B	DX1	COL1	(F)	2474700638101024	V0019020072
B	DX1	COL2	(A)	235350038670109	V0030000073
B	DX1	COL2	(B)	235350070810109	V0012000074
B	DX1	COL2	(C)	2353500838300024	V0001020075
B	DX1	COL3	(A)	223230036390109	V0004000076
B	DX1	COL3	(B)	223230041560109	V0016000077
B	DX1	COL3	(C)	2232300585800024	V0001020078
B		DX2	(A)	346460038420111	H0005050079
B		DX2	(B)	346460044480111	H0005050080
B		DX2	(C)	346460050510111	H0002020081
B		DX2	(D)	346460053550111	H0003030082
B		DX2	(E)	346460057600111	H0004040083
B		DX2	(F)	346460063810111	H0019190084
B		DX2	(G)	334340038670111	H0030300085
B		DX2	(H)	334340070810111	H0012120086
B		DX2	(I)	334340083830011	H0001010087
B		DX2	(J)	322220036390111	H0004040088
B		DX2	(K)	322220041560111	H0016160089

//EOF0544

H02830247NY0051PEDIATRIC

*

H0345

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA 1	217170081790109	V0003000005
B	OTHER UCA 2	2121200817901024V0003010006	
B	DATE (MO)	2434300574601014V0012020007	
B	DATE (DY) 1	1454500555201014V0004010008	
B	DATE (DY) 2	1444400554601014V0010010009	
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014V0013010011	
B	PROV1 CODE	1423901332401014V0040040012	
B	SSN	1332501554601014V0090090013	
B	FMP	1373601554601014V0020020014	
B	VISIT COUNT	2141400544601024V0009010015	
B	PROV1 TIME	2373700351401014V0022020016	
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024V0013010018	
B	PROV2 CODE	1272401332401024V0040040019	
B	PROV2 TIME	2313100351401024V0022020020	
B	BREASON FOR #2 (1)	230300011090109	V0003000021
B	BREASON FOR #2 (2)	2272501080800024H0003010022	
B	IF NOT SCHED	2171700717001024V0002010023	
B	BIF NOT CLINIC (1)	217170063610109	V0003000024
B	BIF NOT CLINIC (2)	2141201606000024H0003010025	
B	REF CODE PFX	2040400706002024V0006010026	
B	REF CODE 1	0030300716001024V0012010027	
B	REF CODE 2	0020101714501024V0054020028	
F	JOB RELATED	801	010029
F	MIL ONLY DUTY	801	010030
F	MIL ONLY QTRS	801	010031
F	MIL ONLY PROF	801	010032
F	FILLER	805	050033
B	SPEC PRE CLIN	306060022140111	H0009090034
B	NOT AVAILABLE	320200009070111	H0003030035
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010036
B	ADDL PROC1	1110701071601024V0050050037	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010038
B	ADDL PROC2	1050101071601024V0050050039	
B	ADMITTED	223230024240011	V0001010040
B	BUNL REAS PRIM PFX	2121200747602024V0002010041	
B	UNL REAS PRIM	1110701748301034V0050050042	
B	UNL REAS SEC PFX	2060600747602024V0002010043	
B	UNL REAS SEC	1050101748301034V0050050044	
B	FOLLOW/R OUT	2383800292801024V0002010045	
B	PROC PROV1 (A)	347470008140111	H0007070046
B	PROC PROV1 (B)	347470017170011	H0001010047
B	PROC PROV1 (C)	347470020250111	H0006060048
B	PROC PROV1 (D)	335350008200111	H0013130049
B	PROC PROV1 (E)	323230008200111	H0013130050
B	PROC PROV2 (A)	346460008140111	H0007070051

B	PROC	PROV2	(B)	346460017170011	H0001010052
B	PROC	PROV2	(C)	346460020250111	H0006060053
B	PROC	PROV2	(D)	334340008200111	H0013130054
B	PROC	PROV2	(E)	322220008200111	H0013130055
B	DX1	COL1	(A)	2474700335701024	V0025020056
B	DX1	COL2	(A)	235350031510109	V0021000057
B	DX1	COL2	(B)	235350054580109	V0005000058
B	DX1	COL2	(C)	235350061640109	V0004000059
B	DX1	COL2	(D)	2353500678101024	V0015020060
B	DX1	COL3	(A)	223230031320109	V0002000061
B	DX1	COL3	(B)	223230035430109	V0009000062
B	DX1	COL3	(C)	223230046540109	V0009000063
B	DX1	COL3	(D)	223230057650109	V0009000064
B	DX1	COL3	(E)	223230068690109	V0002000065
BDX1	COL3	(F)		223230071710009	V0001000066
BDX1	COL3	(G)		223230073730009	V0001000067
BDX1	COL3	(H)		223230075750009	V0001000068
BDX1	COL3	(I)		2232300777901024	V0003010069
BDX1	COL4	(A)		211110031340109	V0004000070
BDX1	COL4	(B)		211110038410109	V0004000071
BDX1	COL4	(C)		211110044450109	V0002000072
BDX1	COL4	(D)		211110047510109	V0005000073
BDX1	COL4	(E)		211110053540109	V0002000074
BDX1	COL4	(F)		211110057580109	V0002000075
BDX1	COL4	(G)		211110061640109	V0004000076
BDX1	COL4	(H)		2111100666701024	V0002020077
B	DX2	(A)		346460033570111	H0025250078
B	DX2	(B)		334340031510111	H0021210079
B	DX2	(C)		334340054580111	H0005050080
B	DX2	(D)		334340061640111	H0004040081
B	DX2	(E)		334340067810111	H0015150082
B	DX2	(F)		322220031320111	H0002020083
B	DX2	(G)		322220035430111	H0009090084
B	DX2	(H)		322220046540111	H0009090085
B	DX2	(I)		322220057650111	H0009090086
B	DX2	(J)		322220068690111	H0002020087
B	DX2	(K)		322220071710011	H0001010088
B	DX2	(L)		322220073730011	H0001010089
B	DX2	(M)		322220075750011	H0001010090
B	DX2	(N)		322220077790111	H0003030091
B	DX2	(O)		310100031340111	H0004040092
B	DX2	(P)		310100038410111	H0004040093
B	DX2	(Q)		310100044450111	H0002020094
B	DX2	(R)		310100047510111	H0005050095
B	DX2	(S)		310100053540111	H0002020096
B	DX2	(T)		310100057580111	H0002020097
B	DX2	(U)		310100061640111	H0004040098

//EOF0552

H03830247NY0051PHYSICAL THERAPY *

H010221 *

B	LITHO	739160104040011	H0024080001
F	UCA PFX	801	010002
F	UCA CODE 1	801	010003
F	UCA CODE 2	802	020004
B	OTHER UCA 1	217170081810009	V0001000005
B	OTHER UCA 2	2121200818100024	V0001010006
B	DATE (MO)	2434300574601014	V0012020007
B	DATE (DY)1	1454500555201014	V0004010008
B	DATE (DY)2	1444400554601014	V0010010009
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014	V0013010011
B	PROV1 CODE	1423901332401014	V0040040012
B	SSN	1332501554601014	V0090090013
B	FMP	1373601554601014	V0020020014
B	VISIT COUNT	2141400544601024	V0009010015
B	PROV1 TIME	2373700351401014	V0022020016
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024	V0013010018
B	PROV2 CODE	1272401332401024	V0040040019
B	PROV2 TIME	2313100351401024	V0022020020
B	BREASON FOR #2 (1)	230300011090109	V0003000021
B	BREASON FOR #2 (2)	227250108080009	H0003000022
B	BREASON FOR #2 (CO)	2303000121200024	V0001010023
B	IF NOT SCHED	2171700717001024	V0002010024
B	BIF NOT CLINIC (1)	217170063610109	V0003000025
B	BIF NOT CLINIC (2)	2141201606000024	H0003010026
B	REF CODE PFX	2040400706002024	V0006010027
B	REF CODE 1	0030300716001024	V0012010028
B	REF CODE 2	0020101714501024	V0054020029
B	JOB RELATED	217170032320011	V0001010030
B	MIL ONLY DUTY	2202000241410024	V0002010031
B	MIL ONLY QTRS	2191900222001024	V0003010032
B	MIL ONLY PROF	2191900181501024	V0004010033
B	REF CODE SING	2090900704501024	V0026020034
F	FILLER	803	030035
B	SPEC PRE CLIN	306060022140111	H0009090036
B	NOT AVAILABLE	320200009070111	H0003030037
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010038
B	ADDL PROC1	1110701071601024	V0050050039
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010040
B	ADDL PROC2	1050101071601024	V0050050041
F	ADMITTED	801	010042
B	BUNL REAS PRIM PFX	2121200747602024	V0002010043
B	UNL REAS PRIM	1110701748301034	V0050050044
B	UNL REAS SEC PFX	2060600747602024	V0002010045
B	UNL REAS SEC	1050101748301034	V0050050046
B	FOLLOW/R OUT	2383800393801024	V0002010047
B	PROC PROV1 (A)	335350008200111	H0013130048
B	PROC PROV1 (B)	335350023300111	H0008080049
B	PROC PROV1 (C)	323230008130111	H0006060050
B	PROC PROV1 (D)	323230015350111	H0021210051

B	PROC PROV1 (E)	311110019350111	H0017170052
B	PROC PROV2 (A)	334340008200111	H0013130053
B	PROC PROV2 (B)	334340023300111	H0008080054
B	PROC PROV2 (C)	322220008130111	H0006060055
B	PROC PROV2 (D)	322220015350111	H0021210056
B	PROC PROV2 (E)	310100019350111	H0017170057
B	OTHER CODES	347470006240111	H0019190058
B	DX1 COL1	2474700437901024	V0037020059
B	DX1 COL2 (A)	235350041490109	V0009000060
B	DX1 COL2 (B)	235350052540109	V0003000061
B	DX1 COL2 (C)	235350056650109	V0010000062
B	DX1 COL2 (D)	235350067680109	V0002000063
B	DX1 COL2 (E)	2353500718001024	V0010020064
B	DX1 COL3	2232300415701024	V0017020065
BDX1	COL4 (A)	211110041460109	V0006000066
BDX1	COL4 (B)	211110049490009	V0001000067
BDX1	COL4 (C)	211110052530109	V0002000068
BDX1	COL4 (D)	211110056560009	V0001000069
BDX1	COL4 (E)	2111100585901024	V0002010070
B	DX2 (A)	346460043790111	H0037370071
B	DX2 (B)	334340041490111	H0009090072
B	DX2 (C)	334340052540111	H0003030073
B	DX2 (D)	334340056650111	H0010100074
B	DX2 (E)	334340067680111	H0002020075
B	DX2 (F)	334340071800111	H0010100076
B	DX2 (G)	322220041570111	H0017170077
B	DX2 (H)	310100041460111	H0006060078
B	DX2 (I)	310100049490011	H0001010079
B	DX2 (J)	310100052530111	H0002020080
B	DX2 (K)	310100056560011	H0001010081

//EOF0593

690 PLASTIC SURGERY

SCAN690.FIL

H03830247NY0051PLASTIC SURGERY *

H010241 *

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY) 1	1454500555201014V0004010007	
B	DATE (DY) 2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	223230008080011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800272601024V0002010044	
B	PROC PROV1 (A)	347470008100111	H0003030045
B	PROC PROV1 (B)	347470012130111	H0002020046
B	PROC PROV1 (C)	347470015150011	H0001010047
B	PROC PROV1 (D)	347470018190111	H0002020048
B	PROC PROV1 (E)	347470021210011	H0001010049
B	PROC PROV1 (F)	335350008100111	H0003030050
B	PROC PROV1 (G)	335350012210111	H0010100051

B	PROC	PROV2	(A)	346460008100111	H0003030052
B	PROC	PROV2	(B)	346460012130111	H0002020053
B	PROC	PROV2	(C)	346460015150011	H0001010054
B	PROC	PROV2	(D)	346460018190111	H0002020055
B	PROC	PROV2	(E)	346460021210011	H0001010056
B	PROC	PROV2	(F)	334340008100111	H0003030057
B	PROC	PROV2	(G)	334340012210111	H0010100058
B	DX1	COL1	(A)	247470032330109	V0002000059
B	DX1	COL1	(B)	247470035370109	V0003000060
B	DX1	COL1	(C)	247470041470109	V0007000061
B	DX1	COL1	(D)	247470050520109	V0003000062
B	DX1	COL1	(E)	247470054570109	V0004000063
B	DX1	COL1	(F)	2474700616301024	V0003020064
B	DX1	COL2	(A)	235350029290009	V0001000065
B	DX1	COL2	(B)	235350031330109	V0003000066
B	DX1	COL2	(C)	235350036410109	V0006000067
B	DX1	COL2	(D)	235350045490109	V0005000068
B	DX1	COL2	(E)	2353500526301024	V0012020069
B	DX1	COL3	(A)	223230029320109	V0004000070
B	DX1	COL3	(B)	223230035350009	V0001000071
B	DX1	COL3	(C)	223230038420109	V0005000072
B	DX1	COL3	(D)	223230044460109	V0003000073
B	DX1	COL3	(E)	223230049550109	V0007000074
B	DX1	COL3	(F)	223230057590109	V0003000075
B	DX1	COL3	(G)	2232300616301024	V0003020076
B	DX1	COL4	(A)	211110029370109	V0009000077
B	DX1	COL4	(B)	211110040450109	V0006000078
B	DX1	COL4	(C)	211110048510109	V0004000079
B	DX1	COL4	(D)	211110053560109	V0004000080
B	DX1	COL4	(E)	211110058590109	V0002000081
B	DX1	COL4	(F)	2111100626200024	V0001020082
B		DX2	(A)	346460032330111	H0002020083
B		DX2	(B)	346460035370111	H0003030084
B		DX2	(C)	346460041470111	H0007070085
B		DX2	(D)	346460050520111	H0003030086
B		DX2	(E)	346460054570111	H0004040087
B		DX2	(F)	346460061630111	H0003030088
B		DX2	(G)	334340029290011	H0001010089
B		DX2	(H)	334340031330111	H0003030090
B		DX2	(I)	334340036410111	H0006060091
B		DX2	(J)	334340045490111	H0005050092
B		DX2	(K)	334340052630111	H0012120093
B		DX2	(L)	322220029320111	H0004040094
B		DX2	(M)	322220035350011	H0001010095
B		DX2	(N)	322220038420111	H0005050096
B		DX2	(O)	322220044460111	H0003030097
B		DX2	(P)	322220049550111	H0007070098
B		DX2	(Q)	322220057590111	H0003030099
B		DX2	(R)	322220061630111	H0003030100
B		DX2	(S)	310100029370111	H0009090101
B		DX2	(T)	310100040450111	H0006060102
B		DX2	(U)	310100048510111	H0004040103
B		DX2	(V)	310100053560111	H0004040104
B		DX2	(W)	310100058590111	H0002020105

//EOF0489

700 PODIATRY

SCAN700.FIL

H03830247NY0051PODIATRY

*

H010234

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
BREASON FOR #2 (1)		230300011090109	V0003000020
BREASON FOR #2 (2)		2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
BIF NOT CLINIC (1)		217170063610109	V0003000023
BIF NOT CLINIC (2)		2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200069070111	H0003030034
BPROV2 ADDL1 (YES)		208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
BPROV2 ADDL2 (YES)		202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	211110027270011	V0001010039
BUNL REAS PRIM PFX		2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800333201024V0002010044	
B	PROC PROV1 (A)	347470008150111	H0008080045
B	PROC PROV1 (B)	347470018240111	H0007070046
B	PROC PROV1 (C)	335350008090111	H0002020047
B	PROC PROV1 (D)	335350011130111	H0003030048
B	PROC PROV1 (E)	335350016230111	H0008080049
B	PROC PROV1 (F)	335350025250011	H0001010050
B	PROC PROV1 (G)	323230008190111	H0012120051

B	PROC	PROV1	(H)	323230022240111	H0003030052
B	PROC	PROV1	(I)	323230027290111	H0003030053
B	PROC	PROV1	(J)	311110020200011	H0001010054
B	PROC	PROV1	(K)	311110022240111	H0003030055
B	PROC	PROV2	(A)	346460008150111	H0008080056
B	PROC	PROV2	(B)	346460018240111	H0007070057
B	PROC	PROV2	(C)	334340008090111	H0002020058
B	PROC	PROV2	(D)	334340011130111	H0003030059
B	PROC	PROV2	(E)	334340016230111	H0008080060
B	PROC	PROV2	(F)	334340025250011	H0001010061
B	PROC	PROV2	(G)	322220008190111	H0012120062
B	PROC	PROV2	(H)	322220022240111	H0003030063
B	PROC	PROV2	(I)	322220027290111	H0003030064
B	PROC	PROV2	(J)	310100020200011	H0001010065
B	PROC	PROV2	(K)	310100022240111	H0003030066
B	DX1	COL1	(A)	247470037510109	V0015000067
B	DX1	COL1	(B)	247470053530009	V0001000068
B	DX1	COL1	(C)	2474700555601024	V0002020069
B	DX1	COL2	(A)	235350035500109	V0016000070
B	DX1	COL2	(B)	2353500527701024	V0026020071
B	DX1	COL3	(A)	223230035440109	V0010000072
B	DX1	COL3	(B)	223230046480109	V0003000073
B	DX1	COL3	(C)	223230052640109	V0013000074
B	DX1	COL3	(D)	223230066710109	V0006000075
B	DX1	COL3	(E)	211110035350009	V0001000076
B	DX1	COL3	(F)	211110037410109	V0005000077
B	DX1	COL3	(G)	2111100434300024	V0001020078
B	DX2	(A)		346460037510111	H0015150079
B	DX2	(B)		346460053530011	H0001010080
B	DX2	(C)		346460055560111	H0002020081
B	DX2	(D)		334340035500111	H0016160082
B	DX2	(E)		334340052770111	H0026260083
B	DX2	(F)		322220035440111	H0010100084
B	DX2	(G)		322220046480111	H0003030085
B	DX2	(H)		322220052640111	H0013130086
B	DX2	(I)		322220066710111	H0006060087
B	DX2	(J)		310100035350011	H0001010088
B	DX2	(K)		310100037410111	H0005050089

//EOF0545

710 PREVENTIVE MEDICINE

SCAN710.FIL

H03830247NY0051PREVENTIVE MEDICINE *

H010229 *

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA 1	217170081810009	V0001000005
B	OTHER UCA 2	2121200818100024V0001010006	
B	DATE (MO)	2434300574601014V0012020007	
B	DATE (DY)1	1454500555201014V0004010008	
B	DATE (DY)2	1444400554601014V0010010009	
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014V0013010011	
B	PROV1 CODE	1423901332401014V0040040012	
B	SSN	1332501554601014V0090090013	
B	FMP	1373601554601014V0020020014	
B	VISIT COUNT	2141400544601024V0009010015	
B	PROV1 TIME	2373700351401014V0022020016	
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024V0013010018	
B	PROV2 CODE	1272401332401024V0040040019	
B	PROV2 TIME	2313100351401024V0022020020	
B	BREASON FOR #2 (1)	230300011090109	V0003000021
B	BREASON FOR #2 (2)	2272501080800024H0003010022	
B	IF NOT SCHED	2171700717001024V0002010023	
B	BIF NOT CLINIC (1)	217170063610109	V0003000024
B	BIF NOT CLINIC (2)	2141201606000024H0003010025	
B	REF CODE PFX	2040400706002024V0006010026	
B	REF CODE 1	0030300716001024V0012010027	
B	REF CODE 2	0020101714501024V0054020028	
B	JOB RELATED	217170032320011	V0001010029
B	MIL ONLY DUTY	2202000241410024V0002010030	
B	MIL ONLY QTRS	2191900222001024V0003010031	
B	MIL ONLY PROF	2191900181501024V0004010032	
F	FILLER	805	050033
B	SPEC PRE CLIN	306060022140111	H0009090034
B	NOT AVAILABLE	320200009070111	H0003030035
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010036
B	ADDL PROC1	1110701071601024V0050050037	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010038
B	ADDL PROC2	1050101071601024V0050050039	
F	ADMITTED	801	010040
B	BUNL REAS PRIM PFX	2121200747602024V0002010041	
B	UNL REAS PRIM	1110701748301034V0050050042	
B	UNL REAS SEC PFX	2060600747602024V0002010043	
B	UNL REAS SEC	1050101748301034V0050050044	
B	FOLLOW/R OUT	2383800363501024V0002010045	
B	PROC PROV1 (A)	347470008170111	H0010100046
B	PROC PROV1 (B)	347470020290111	H0010100047
B	PROC PROV1 (C)	335350008100111	H0003030048
B	PROC PROV1 (D)	335350013320111	H0020200049
B	PROC PROV1 (E)	323230008140111	H0007070050
B	PROC PROV1 (F)	323230017220111	H0006060051

B	PROC	PROV2	(A)	346460008170111	H0010100052
B	PROC	PROV2	(B)	346460020290111	H0010100053
B	PROC	PROV2	(C)	334340008100111	H0003030054
B	PROC	PROV2	(D)	334340013320111	H0020200055
B	PROC	PROV2	(E)	322220008140111	H0007070056
B	PROC	PROV2	(F)	322220017220111	H0006060057
BDX1	COL1		(A)	247470040490109	V0010000058
BDX1	COL1		(B)	247470052580109	V0007000059
BDX1	COL1		(C)	247470060680109	V0009000060
BDX1	COL1		(D)	247470071720109	V0002000061
BDX1	COL1		(E)	247470075760109	V0002000062
BDX1	COL1		(F)	2474700798001024	V0002020063
BDX1	COL2		(A)	235350038390109	V0002000064
BDX1	COL2		(B)	235350043460109	V0004000065
BDX1	COL2		(C)	235350049530109	V0005000066
BDX1	COL2		(D)	235350056630109	V0008000067
BDX1	COL2		(E)	235350066700109	V0005000068
BDX1	COL2		(F)	235350073760109	V0004000069
BDX1	COL2		(G)	2353500798101024	V0003020070
BDX1	COL3		(A)	223230038420109	V0005000071
BDX1	COL3		(B)	223230045490109	V0005000072
BDX1	COL3		(C)	223230052730109	V0022000073
BDX1	COL3		(D)	223230075750009	V0001000074
BDX1	COL3		(E)	2232300777700024	V0001020075
BDX1	COL4		(A)	211110037370009	V0001000076
BDX1	COL4		(B)	211110039390009	V0001000077
BDX1	COL4		(C)	211110041410009	V0001000078
BDX1	COL4		(D)	211110043450109	V0003000079
BDX1	COL4		(E)	211110047470009	V0001000080
BDX1	COL4		(F)	211110049550109	V0007000081
BDX1	COL4		(G)	211110058610109	V0004000082
BDX1	COL4		(H)	2111100636401024	V0002020083
B	DX2		(A)	346460040490111	H0010100084
B	DX2		(B)	346460052580111	H0007070085
B	DX2		(C)	346460060680111	H0009090086
B	DX2		(D)	346460071720111	H0002020087
B	DX2		(E)	346460075760111	H0002020088
B	DX2		(F)	346460079800111	H0002020089
B	DX2		(G)	334340038390111	H0002020090
B	DX2		(H)	334340043460111	H0004040091
B	DX2		(I)	334340049530111	H0005050092
B	DX2		(J)	334340056630111	H0008080093
B	DX2		(K)	334340066700111	H0005050094
B	DX2		(L)	334340073760111	H0004040095
B	DX2		(M)	334340079810111	H0003030096
B	DX2		(N)	322220038420111	H0005050097
B	DX2		(O)	322220045490111	H0005050098
B	DX2		(P)	322220052730111	H0022220099
B	DX2		(Q+)	322220075750011	H0001010100
B	DX2		(R)	322220077770011	H0001010101
B	DX2		(S)	310100037370011	H0001010102
B	DX2		(T)	310100039390011	H0001010103
B	DX2		(U)	310100041410011	H0001010104
B	DX2		(V)	310100043450111	H0003030105
B	DX2		(W)	310100047470011	H0001010106

B DX2 (X) 310100049550111 H0007070107
B DX2 (Y) 310100058610111 H0004040108
//EOF0572

720 PRIMARY CARE

SCAN720.FIL

H03830247NY0051PRIMARY CARE *

H010240 *

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
B	OTHER UCA	217170081810009	V0001000005
B	OTHER UCA 1	2121200818100024	V0001010006
B	DATE (MO)	2434300574601014	V0012020007
B	DATE (DY)1	1454500555201014	V0004010008
B	DATE (DY)2	1444400554601014	V0010010009
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014	V0013010011
B	PROV1 CODE	1423901332401014	V0040040012
B	SSN	1332501554601014	V0090090013
B	FMP	1373601554601014	V0020020014
B	VISIT COUNT	2141400544601024	V0009010015
B	PROV1 TIME	2373700351401014	V0022020016
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024	V0013010018
B	PROV2 CODE	1272401332401024	V0040040019
B	PROV2 TIME	2313100351401024	V0022020020
B	BREASON FOR #2 (1)	230300011090109	V0003000021
B	BREASON FOR #2 (2)	2272501080800024	H0003010022
B	IF NOT SCHED	2171700717001024	V0002010023
B	BIF NOT CLINIC (1)	217170063610109	V0003000024
B	BIF NOT CLINIC (2)	2141201606000024	H0003010025
B	REF CODE PFX	2040400706002024	V0006010026
B	REF CODE 1	0030300716001024	V0012010027
B	REF CODE 2	0020101714501024	V0054020028
B	JOB RELATED	217170032320011	V0001010029
B	MIL ONLY DUTY	2202000241410024	V0002010030
B	MIL ONLY QTRS	2191900222001024	V0003010031
B	MIL ONLY PROF	2191900181501024	V0004010032
F	FILLER	805	050033
B	SPEC PRE CLIN	306060022140111	H0009090034
B	NOT AVAILABLE	320200009070111	H0003030035
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010036
B	ADDL PROC1	1110701071601024	V0050050037
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010038
B	ADDL PROC2	1050101071601024	V0050050039
B	ADMITTED	247470020200011	V0001010040
B	BUNL REAS PRIM PFX	2121200747602024	V0002010041
B	UNL REAS PRIM	1110701748301034	V0050050042
B	UNL REAS SEC PFX	2060600747602024	V0002010043
B	UNL REAS SEC	1050101748301034	V0050050044
B	FOLLOW/R OUT	2383800272601024	V0002010045
B	PROC PROV1	347470007180111	H0012120046
B	PROC PROV2	346460007180111	H0012120047
B	DX1 COL1 (A)	247470031350109	V0005000048
B	DX1 COL1 (B)	247470038410109	V0004000049
B	DX1 COL1 (C)	247470043450109	V0003000050
B	DX1 COL1 (D)	247470047490109	V0003000051

B	DX1	COL1	(E)	247470051530109	V0003000052
B	DX1	COL1	(F)	247470056610109	V0006000053
B	DX1	COL1	(G)	2474700647301024	V0010020054
B	DX1	COL2	(A)	235350030310109	V0002000055
B	DX1	COL2	(B)	235350033350109	V0003000056
B	DX1	COL2	(C)	235350038440109	V0007000057
B	DX1	COL2	(D)	235350047500109	V0004000058
B	DX1	COL2	(E)	2353500536501024	V0013020059
B	DX1	COL3	(A)	223230029340109	V0006000060
B	DX1	COL3	(B)	223230037410109	V0005000061
B	DX1	COL3	(C)	223230044450109	V0002000062
B	DX1	COL3	(D)	223230049490009	V0001000063
B	DX1	COL3	(E)	223230051540109	V0004000064
B	DX1	COL3	(F)	223230056620109	V0007000065
B	DX1	COL3	(G)	2232300646601024	V0003020066
B	DX1	COL4	(A)	211110029290009	V0001000067
B	DX1	COL4	(B)	211110031310009	V0001000068
B	DX1	COL4	(C)	211110033330009	V0001000069
B	DX1	COL4	(D)	211110036360009	V0001000070
B	DX1	COL4	(E)	211110038380009	V0001000071
B	DX1	COL4	(F)	211110041430109	V0003000072
B	DX1	COL4	(G)	211110045460109	V0002000073
B	DX1	COL4	(H)	211110049510109	V0003000074
B	DX1	COL4	(I)	2111100535401024	V0002020075
B	DX2	(A)		346460031350111	H0005050076
B	DX2	(B)		346460038410111	H0004040077
B	DX2	(C)		346460043450111	H0003030078
B	DX2	(D)		346460047490111	H0003030079
B	DX2	(E)		346460051530111	H0003030080
B	DX2	(F)		346460056610111	H0006060081
B	DX2	(G)		346460064730111	H0010100082
B	DX2	(H)		334340030310111	H0002020083
B	DX2	(I)		334340033350111	H0003030084
B	DX2	(J)		334340038440111	H0007070085
B	DX2	(K)		334340047500111	H0004040086
B	DX2	(L)		334340053650111	H0013130087
B	DX2	(M)		322220029340111	H0006060088
B	DX2	(N)		322220037410111	H0005050089
B	DX2	(O)		322220044450111	H0002020090
B	DX2	(P)		322220049490011	H0001010091
B	DX2	(Q)		322220051540111	H0004040092
B	DX2	(R)		322220056620111	H0007070093
B	DX2	(S)		322220064660111	H0003030094
B	DX2	(T)		310100029290011	H0001010095
B	DX2	(U)		310100031310011	H0001010096
B	DX2	(V)		310100033330011	H0001010097
B	DX2	(W)		310100036360011	H0001010098
B	DX2	(X)		310100038380011	H0001010099
B	DX2	(Y)		310100041430111	H0003030100
B	DX2	(Z)		310100045460111	H0002020101
B	DX2	(AA)		310100049510111	H0003030102

//EOF0473

730 PSYCHIATRY

SCAN730.FIL

H03830247NY0051PSYCHIATRY

*

H020444

*

B LITHO 739160104040011 H0024080001
B UCA PFX 2090900706202024V0005010002
B UCA CODE 1 0080800716001024V0012010003
B UCA CODE 2 0070601714501024V0054020004
B OTHER UCA 1 217170081810009 V0001000005
B OTHER UCA 2 2121200818100024V0001010006
B DATE (MO) 2434300574601014V0012020007
B DATE (DY)1 1454500555201014V0004010008
B DATE (DY)2 1444400554601014V0010010009
B PROV1 PFX1 044440033200109 V0014000010
B PROV1 PFX2 0434300322001014V0013010011
B PROV1 CODE 1423901332401014V0040040012
B SSN 1332501554601014V0090090013
B FMP 1373601554601014V0020020014
B VISIT COUNT 2141400544601024V0009010015
B PROV1 TIME 2373700351401014V0022020016
B PROV2 PFX1 029290033200109 V0014000017
B PROV2 PFX2 0282800322001024V0013010018
B PROV2 CODE 1272401332401024V0040040019
B PROV2 TIME 2313100351401024V0022020020
BREASON FOR #2 (1) 230300011090109 V0003000021
BREASON FOR #2 (2) 227250108080009 H0003000022
BREASON FOR #2 (CO) 2303000121200024V0001010023
B IF NOT SCHED 2171700717001024V0002010024
BIF NOT CLINIC (1) 217170063610109 V0003000025
BIF NOT CLINIC (2) 2141201606000024H0003010026
B REF CODE PFX 2040400706002024V0006010027
B REF CODE 1 0030300716001024V0012010028
B REF CODE 2 0020101714501024V0054020029
B JOB RELATED 217170032320011 V0001010030
B MIL ONLY DUTY 2202000241410024V0002010031
B MIL ONLY QTRS 2191900222001024V0003010032
B MIL ONLY PROF 2191900181501024V0004010033
F FILLER 805 050034
B SPEC PRE CLIN 306060022140111 H0009090035
B NOT AVAILABLE 320200009070111 H0003030036
BPROV2 ADDL1 (YES) 208080006060011 V0001010037
B ADDL PROC1 1110701071601024V0050050038
BPROV2 ADDL2 (YES) 202020006060011 V0001010039
B ADDL PROC2 1050101071601024V0050050040
B ADMITTED 211110017170011 V0001010041
BUNL REAS PRIM PFX 2121200747602024V0002010042
B UNL REAS PRIM 1110701748301034V0050050043
B UNL REAS SEC PFX 2060600747602024V0002010044
B UNL REAS SEC 1050101748301034V0050050045
B RULE OUT 244440005050011 V0001010046
B PROC PROV1 323230007160111 H0010100047
B PROC PROV2 322220007160111 H0010100048
BOTHER CODES (AII) 311110036480111 H0013130049
BOTHER CODES (AIII) 311110050500011 H0001010050
BOTHER CODES (AIV) 311110052590111 H0008080051

BOTHER CODES (AV)	311110061680111	H0008080052
B AXIS1 COL1 (A)	247470007080109	V0002000053
B AXIS1 COL1 (B)	247470011120109	V0002000054
B AXIS1 COL1 (C)	247470014170109	V0004000055
B AXIS1 COL1 (D)	247470019240109	V0006000056
B AXIS1 COL1 (E)	247470026270109	V0002000057
B AXIS1 COL1 (F)	247470029300109	V0002000058
B AXIS1 COL1 (G)	247470032360109	V0005000059
B AXIS1 COL1 (H)	247470038390109	V0002000060
B AXIS1 COL1 (I)	247470042470109	V0006000061
B AXIS1 COL1 (J)	247470051560109	V0006000062
B AXIS1 COL1 (K)	247470058640109	V0007000063
B AXIS1 COL1 (L)	247470066690109	V0004000064
B AXIS1 COL1 (M)	247470071780109	V0008000065
B AXIS1 COL1 (N)	2474700808301024	V0004020066
B AXIS1 COL2 (A)	235350006100109	V0005000067
B AXIS1 COL2 (B)	235350012290109	V0018000068
B AXIS1 COL2 (C)	235350033400109	V0008000069
B AXIS1 COL2 (D)	235350042460109	V0005000070
B AXIS1 COL2 (E)	235350053550109	V0003000071
B AXIS1 COL2 (F)	235350057600109	V0004000072
B AXIS1 COL2 (G)	235350062640109	V0003000073
B AXIS1 COL2 (H)	235350070710109	V0002000074
B AXIS1 COL2 (I)	235350079810109	V0003000075
B AXIS1 COL2 (J)	2353500838300024	V0001020076
B AXIS1 COL3 (A)	223230018260109	V0009000077
B AXIS1 COL3 (B)	223230028310109	V0004000078
B AXIS1 COL3 (C)	223230033360109	V0004000079
B AXIS1 COL3 (D)	223230038380009	V0001000080
B AXIS1 COL3 (E)	223230041410009	V0001000081
B AXIS1 COL3 (F)	223230043510109	V0009000082
B AXIS1 COL3 (G)	223230053600109	V0008000083
B AXIS1 COL3 (H)	223230062630109	V0002000084
B AXIS1 COL3 (I)	223230065700109	V0006000085
B AXIS1 COL3 (J)	223230072800109	V0009000086
B AXIS1 COL3 (K)	2232300828301024	V0002020087
B AXIS1 COL4	2111100183101024	V0014020088
B ADDL AXIS1 (A)	346460007080111	H0002020089
B ADDL AXIS1 (B)	346460011120111	H0002020090
B ADDL AXIS1 (C)	346460014170111	H0004040091
B ADDL AXIS1 (D)	346460019240111	H0006060092
B ADDL AXIS1 (E)	346460026270111	H0002020093
B ADDL AXIS1 (F)	346460029300111	H0002020094
B ADDL AXIS1 (G)	346460032360111	H0005050095
B ADDL AXIS1 (H)	346460038390111	H0002020096
B ADDL AXIS1 (I)	346460042470111	H0006060097
B ADDL AXIS1 (J)	346460051560111	H0006060098
B ADDL AXIS1 (K)	346460058640111	H0007070099
B ADDL AXIS1 (L)	346460066690111	H0004040100
B ADDL AXIS1 (M)	346460071780111	H0008080101
B ADDL AXIS1 (N)	346460080830111	H0004040102
B ADDL AXIS1 (O)	334340006100111	H0005050103
B ADDL AXIS1 (P)	334340012290111	H0018180104
B ADDL AXIS1 (Q)	334340033400111	H0008080105
B ADDL AXIS1 (R)	334340042460111	H0005050106

B	ADDL	AXIS1	(S)	334340053550111	H0003030107
B	ADDL	AXIS1	(T)	334340057600111	H0004040108
B	ADDL	AXIS1	(U)	334340062640111	H0003030109
B	ADDL	AXIS1	(V)	334340070710111	H0002020110
B	ADDL	AXIS1	(W)	334340079810111	H0003030111
B	ADDL	AXIS1	(X)	334340083830011	H0001010112
B	ADDL	AXIS1	(Y)	322220018260111	H0009090113
B	ADDL	AXIS1	(Z)	322220028310111	H0004040114
B	ADDL	AXIS1	(AA)	322220033360111	H0004040115
B	ADDL	AXIS1	(BB)	322220038380011	H0001010116
B	ADDL	AXIS1	(CC)	322220041410011	H0001010117
B	ADDL	AXIS1	(DD)	322220043510111	H0009090118
B	ADDL	AXIS1	(EE)	322220053600111	H0008080119
B	ADDL	AXIS1	(FF)	322220062630111	H0002020120
B	ADDL	AXIS1	(GG)	322220065700111	H0006060121
B	ADDL	AXIS1	(HH)	322220072800111	H0009090122
B	ADDL	AXIS1	(II)	322220082830111	H0002020123
B	ADDL	AXIS1	(JJ)	310100018290111	H0012120124
B	SPEC	PROG	(1)	307070035320111	H0004040125
B	SPEC	PROG	(2)	304020131310011	V0003030126

//EOF0574

H03830247NY0051PSYCHOLOGY

*

H020443

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA 1	217170081800109	V0002000005
B	OTHER UCA 2	2121200818100024V0001010006	
B	DATE (MO)	2434300574601014V0012020007	
B	DATE (DY) 1	1454500555201014V0004010008	
B	DATE (DY) 2	1444400554601014V0010010009	
B	PROV1 PFX1	044440033200109	V0014000010
B	PROV1 PFX2	0434300322001014V0013010011	
B	PROV1 CODE	1423901332401014V0040040012	
B	SSN	1332501554601014V0090090013	
B	FMP	1373601554601014V0020020014	
B	VISIT COUNT	2141400544601024V0009010015	
B	PROV1 TIME	2373700351401014V0022020016	
B	PROV2 PFX1	029290033200109	V0014000017
B	PROV2 PFX2	0282800322001024V0013010018	
B	PROV2 CODE	1272401332401024V0040040019	
B	PROV2 TIME	2313100351401024V0022020020	
B	BREASON FOR #2 (1)	230300011090109	V0003000021
B	BREASON FOR #2 (2)	227250108080009	H0003000022
B	BREASON FOR #2 (CO)	2303000121200024V0001010023	
B	IF NOT SCHED	2171700717001024V0002010024	
B	BIF NOT CLINIC (1)	217170063610109	V0003000025
B	BIF NOT CLINIC (2)	2141201606000024H0003010026	
B	REF CODE PFX	2040400706002024V0006010027	
B	REF CODE 1	0030300716001024V0012010028	
B	REF CODE 2	0020101714501024V0054020029	
B	JOB RELATED	217170032320011	V0001010030
B	MIL ONLY DUTY	2202000241410024V0002010031	
B	MIL ONLY QTRS	2191900222001024V0003010032	
B	MIL ONLY PROF	2191900181501024V0004010033	
F	FILLER	805	050034
B	SPEC PRE CLIN	306060022140111	H0009090035
B	NOT AVAILABLE	320200009070111	H0003030036
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010037
B	ADDL PROC1	1110701071601024V0050050038	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010039
B	ADDL PROC2	1050101071601024V0050050040	
F	ADMITTED	801	010041
B	BUNL REAS PRIM PFX	2121200747602024V0002010042	
B	UNL REAS PRIM	1110701748301034V0050050043	
B	UNL REAS SEC PFX	2060600747602024V0002010044	
B	UNL REAS SEC	1050101748301034V0050050045	
B	RULE OUT	242420022220011	V0001010046
B	PROC PROV1 (A)	347470007190111	H0013130047
B	PROC PROV1 (B)	335350006100111	H0005050048
B	PROC PROV1 (C)	335350014190111	H0006060049
B	PROC PROV1 (D)	323230006190111	H0014140050
B	PROC PROV2 (A)	346460007190111	H0013130051

B	PROC PROV2	(B)	334340006100111	H0005050052
B	PROC PROV2	(C)	334340014190111	H0006060053
B	PROC PROV2	(D)	322220006190111	H0014140054
B	BOTHER CODES	(AII)	1311110031330111	H0003030055
B	BOTHER CODES	(AII)	2311110035480111	H0014140056
B	BOTHER CODES	(AIII)	311110050500011	H0001010057
B	BOTHER CODES	(AIV)	311110052590111	H0008080058
B	BOTHER CODES	(AV)	311110061680111	H0008080059
B	AXIS1 COL1	(A)	247470024290109	V0006000060
B	AXIS1 COL1	(B)	247470031330109	V0003000061
B	AXIS1 COL1	(C)	247470035390109	V0005000062
B	AXIS1 COL1	(D)	247470041540109	V0014000063
B	AXIS1 COL1	(E)	247470056610109	V0006000064
B	AXIS1 COL1	(F)	247470065740109	V0010000065
B	AXIS1 COL1	(G)	2474700768301024	V0008020066
B	AXIS1 COL2	(A)	235350024400109	V0017000067
B	AXIS1 COL2	(B)	235350044480109	V0005000068
B	AXIS1 COL2	(C)	235350054560109	V0003000069
B	AXIS1 COL2	(D)	235350058610109	V0004000070
B	AXIS1 COL2	(E)	235350063650109	V0003000071
B	AXIS1 COL2	(F)	235350070710109	V0002000072
B	AXIS1 COL2	(G)	235350077800109	V0004000073
B	AXIS1 COL2	(H)	2353500828301024	V0002020074
B	AXIS1 COL3	(A)	223230023350109	V0013000075
B	AXIS1 COL3	(B)	223230037400109	V0004000076
B	AXIS1 COL3	(C)	223230042510109	V0010000077
B	AXIS1 COL3	(D)	223230053550109	V0003000078
B	AXIS1 COL3	(E)	223230057600109	V0004000079
B	AXIS1 COL3	(F)	223230062700109	V0009000080
B	AXIS1 COL3	(G)	223230072830109	V0012000081
B	AXIS1 COL3	(H)	2111100232601024	V0004020082
B	ADDL AXIS1	(A)	346460024290111	H0006060083
B	ADDL AXIS1	(B)	346460031330111	H0003030084
B	ADDL AXIS1	(C)	346460035390111	H0005050085
B	ADDL AXIS1	(D)	346460041540111	H0014140086
B	ADDL AXIS1	(E)	346460056610111	H0006060087
B	ADDL AXIS1	(F)	346460065740111	H0010100088
B	ADDL AXIS1	(G)	346460076830111	H0008080089
B	ADDL AXIS1	(H)	334340024400111	H0017170090
B	ADDL AXIS1	(I)	334340044480111	H0005050091
B	ADDL AXIS1	(J)	334340054560111	H0003030092
B	ADDL AXIS1	(K)	334340058610111	H0004040093
B	ADDL AXIS1	(L)	334340063650111	H0003030094
B	ADDL AXIS1	(M)	334340070710111	H0002020095
B	ADDL AXIS1	(N)	334340077800111	H0004040096
B	ADDL AXIS1	(O)	334340082830111	H0002020097
B	ADDL AXIS1	(P)	322220023350111	H0013130098
B	ADDL AXIS1	(Q)	322220037400111	H0004040099
B	ADDL AXIS1	(R)	322220042510111	H0010100100
B	ADDL AXIS1	(S)	322220053550111	H0003030101
B	ADDL AXIS1	(T)	322220057600111	H0004040102
B	ADDL AXIS1	(U)	322220062700111	H0009090103
B	ADDL AXIS1	(V)	322220072830111	H0012120104
B	ADDL AXIS1	(W)	310100023260111	H0004040105
B	EPYCHOMETRIC ASSES		311110018210111	H0004040106

B SPEC PROG (1) 307070035320111 H0004040107
B SPEC PROG (2) 304020131310011 V0003030108
//EOF0608

750 PULMONARY

SCAN750.FIL

H03830247NY0051PULMONARY

*

H010230

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
B	OTHER UCA	2171205818100024	H0002010005
B	DATE (MO)	2434300574601014	V0012020006
B	DATE (DY)1	1454500555201014	V0004010007
B	DATE (DY)2	1444400554601014	V0010010008
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014	V0013010010
B	PROV1 CODE	1423901332401014	V0040040011
B	SSN	1332501554601014	V0090090012
B	FMP	1373601554601014	V0020020013
B	VISIT COUNT	2141400544601024	V0009010014
B	PROV1 TIME	2373700351401014	V0022020015
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024	V0013010017
B	PROV2 CODE	1272401332401024	V0040040018
B	PROV2 TIME	2313100351401024	V0022020019
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024	H0003010021
B	IF NOT SCHED	2171700717001024	V0002010022
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024	H0003010024
B	REF CODE PFX	2040400706002024	V0006010025
B	REF CODE 1	0030300716001024	V0012010026
B	REF CODE 2	0020101714501024	V0054020027
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024	V0002010029
B	MIL ONLY QTRS	2191900222001024	V0003010030
B	MIL ONLY PROF	2191900181501024	V0004010031
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024	V0050050036
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024	V0050050038
B	ADMITTED	211110020200011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024	V0002010040
B	UNL REAS PRIM	1110701748301034	V0050050041
B	UNL REAS SEC PFX	2060600747602024	V0002010042
B	UNL REAS SEC	1050101748301034	V0050050043
B	FOLLOW/R OUT	2383800282701024	V0002010044
B	PROC PROV1 (A)	347470009090011	H0001010045
B	PROC PROV1 (B)	347470011130111	H0003030046
B	PROC PROV1 (C)	347470015150011	H0001010047
B	PROC PROV1 (D)	347470018190111	H0002020048
B	PROC PROV1 (E)	347470022240111	H0003030049
B	PROC PROV1 (F)	335350008080011	H0001010050
B	PROC PROV1 (G)	335350012230111	H0012120051

B	PROC	PROV1	(H)	323230009110111	H0003030052
B	PROC	PROV1	(I)	323230013130011	H0001010053
B	PROC	PROV1	(J)	323230015160111	H0002020054
B	PROC	PROV1	(K)	323230018220111	H0005050055
B	PROC	PROV1	(L)	323230024240011	H0001010056
B	PROC	PROV1	(M)	311110018180011	H0001010057
B	PROC	PROV2	(A)	346460009090011	H0001010058
B	PROC	PROV2	(B)	346460011130111	H0003030059
B	PROC	PROV2	(C)	346460015150011	H0001010060
B	PROC	PROV2	(D)	346460018190111	H0002020061
B	PROC	PROV2	(E)	346460022240111	H0003030062
B	PROC	PROV2	(F)	334340008080011	H0001010063
B	PROC	PROV2	(G)	334340012230111	H0012120064
B	PROC	PROV2	(H)	322220009110111	H0003030065
B	PROC	PROV2	(I)	322220013130011	H0001010066
B	PROC	PROV2	(J)	322220015160111	H0002020067
B	PROC	PROV2	(K)	322220018220111	H0005050068
B	PROC	PROV2	(L)	322220024240011	H0001010069
B	PROC	PROV2	(M)	310100018180011	H0001010070
B	DX1	COL1	(A)	247470032390109	V0008000071
B	DX1	COL1	(B)	247470042500109	V0009000072
B	DX1	COL1	(C)	247470053550109	V0003000073
B	DX1	COL1	(D)	247470057620109	V0006000074
B	DX1	COL1	(E)	247470064640009	V0001000075
B	DX1	COL1	(F)	247470066690109	V0004000076
B	DX1	COL1	(G)	247470071730109	V0003000077
B	DX1	COL1	(H)	247470075750009	V0001000078
B	DX1	COL1	(I)	2474700778101024	V0005020079
B	DX1	COL2	(A)	235350030310109	V0002000080
B	DX1	COL2	(B)	235350033360109	V0004000081
B	DX1	COL2	(C)	235350038420109	V0005000082
B	DX1	COL2	(D)	235350045470109	V0003000083
B	DX1	COL2	(E)	235350049490009	V0001000084
B	DX1	COL2	(F)	235350051530109	V0003000085
B	DX1	COL2	(G)	235350055560109	V0002000086
B	DX1	COL2	(H)	235350059600109	V0002000087
B	DX1	COL2	(I)	2353500626401024	V0003020088
B	DX1	COL3	(A)	223230031310009	V0001000089
B	DX1	COL3	(B)	223230033360109	V0004000090
B	DX1	COL3	(C)	223230039410109	V0003000091
B	DX1	COL3	(D)	223230044590109	V0016000092
B	DX1	COL3	(E)	2232300616100024	V0001020093
B	DX2	(A)		346460032390111	H0008080094
B	DX2	(B)		346460042500111	H0009090095
B	DX2	(C)		346460053550111	H0003030096
B	DX2	(D)		346460057620111	H0006060097
B	DX2	(E)		346460064640011	H0001010098
B	DX2	(F)		346460066690111	H0004040099
B	DX2	(G)		346460071730111	H0003030100
B	DX2	(H)		346460075750011	H0001010101
B	DX2	(I)		346460077810111	H0005050102
B	DX2	(J)		334340030310111	H0002020103
B	DX2	(K)		334340033360111	H0004040104
B	DX2	(L)		334340038420111	H0005050105
B	DX2	(M)		334340045470111	H0003030106

B	DX2 (N)	334340049490011	H0001010107
B	DX2 (O)	334340051530111	H0003030108
B	DX2 (P)	334340055560111	H0002020109
B	DX2 (Q)	334340059600111	H0002020110
B	DX2 (R)	334340062640111	H0003030111
B	DX2 (S)	322220031310011	H0001010112
B	DX2 (T)	322220033360111	H0004040113
B	DX2 (U)	322220039410111	H0003030114
B	DX2 (V)	322220044590111	H0016160115

//EOF0504

770 RHEUMATOLOGY

SCAN770.FIL

H03830247NY0051RHEUMATOLOGY

*

H010226

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024	V0005010002
B	UCA CODE 1	0080800716001024	V0012010003
B	UCA CODE 2	0070601714501024	V0054020004
B	OTHER UCA	2171700818100024	V0001010005
B	DATE (MO)	2434300574601014	V0012020006
B	DATE (DY) 1	1454500555201014	V0004010007
B	DATE (DY) 2	1444400554601014	V0010010008
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014	V0013010010
B	PROV1 CODE	1423901332401014	V0040040011
B	SSN	1332501554601014	V0090090012
B	FMP	1373601554601014	V0020020013
B	VISIT COUNT	2141400544601024	V0009010014
B	PROV1 TIME	2373700351401014	V0022020015
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024	V0013010017
B	PROV2 CODE	1272401332401024	V0040040018
B	PROV2 TIME	2313100351401024	V0022020019
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024	H0003010021
B	IF NOT SCHED	2171700717001024	V0002010022
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024	H0003010024
B	REF CODE PFX	2040400706002024	V0006010025
B	REF CODE 1	0030300716001024	V0012010026
B	REF CODE 2	0020101714501024	V0054020027
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024	V0002010029
B	MIL ONLY QTRS	2191900222001024	V0003010030
B	MIL ONLY PROF	2191900181501024	V0004010031
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024	V0050050036
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024	V0050050038
B	ADMITTED	235350021210011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024	V0002010040
B	UNL REAS PRIM	1110701748301034	V0050050041
B	UNL REAS SEC PFX	2060600747602024	V0002010042
B	UNL REAS SEC	1050101748301034	V0050050043
B	FOLLOW/R OUT	2383800272601024	V0002010044
B	PROC PROV1 (A)	347470008150111	H0008080045
B	PROC PROV1 (B)	347470017170011	H0001010046
B	PROC PROV2 (A)	346460008150111	H0008080047
B	PROC PROV2 (B)	346460017170011	H0001010048
B	OTHER CODES	334340009190111	H0011110049
B	DX1 COL1 (A)	247470031370109	V0007000050
B	DX1 COL1 (B)	247470040440109	V0005000051

B	DX1	COL1	(C)	247470048500109	V0003000052
B	DX1	COL1	(D)	247470053530009	V0001000053
B	DX1	COL1	(E)	247470055580109	V0004000054
B	DX1	COL1	(F)	247470060630109	V0004000055
B	DX1	COL1	(G)	247470065710109	V0007000056
B	DX1	COL1	(H)	2474700747801024	V0005020057
B	DX1	COL2	(A)	235350029300109	V0002000058
B	DX1	COL2	(B)	235350033330009	V0001000059
B	DX1	COL2	(C)	235350035390109	V0005000060
B	DX1	COL2	(D)	235350042430109	V0002000061
B	DX1	COL2	(E)	235350046530109	V0008000062
B	DX1	COL2	(F)	235350056620109	V0007000063
B	DX1	COL2	(G)	235350065670109	V0003000064
B	DX1	COL2	(H)	235350070710109	V0002000065
B	DX1	COL2	(I)	2353500737401024	V0002020066
B	DX1	COL3	(A)	223230029300109	V0002000067
B	DX1	COL3	(B)	223230033370109	V0005000068
B	DX1	COL3	(C)	223230041440109	V0004000069
B	DX1	COL3	(D)	223230047470009	V0001000070
B	DX1	COL3	(E)	223230049490009	V0001000071
B	DX1	COL3	(F)	223230053540109	V0002000072
B	DX1	COL3	(G)	223230057610109	V0005000073
B	DX1	COL3	(H)	2232300647701024	V0014020074
B	DX1	COL4	(A)	211110029300109	V0002000075
B	DX1	COL4	(B)	211110032330109	V0002000076
B	DX1	COL4	(C)	211110035360109	V0002000077
B	DX1	COL4	(D)	211110038390109	V0002000078
B	DX1	COL4	(E)	211110041430109	V0003000079
B	DX1	COL4	(F)	211110045460109	V0002000080
B	DX1	COL4	(G)	211110049490009	V0001000081
B	DX1	COL4	(H)	211110051520109	V0002000082
B	DX1	COL4	(I)	211110054640109	V0011000083
B	DX1	COL4	(J)	2111100666600024	V0001020084
B	DX2	(A)		346460031370111	H0007070085
B	DX2	(B)		346460040440111	H0005050086
B	DX2	(C)		346460048500111	H0003030087
B	DX2	(D)		346460053530011	H0001010088
B	DX2	(E)		346460055580111	H0004040089
B	DX2	(F)		346460060630111	H0004040090
B	DX2	(G)		346460065710111	H0007070091
B	DX2	(H)		346460074780111	H0005050092
B	DX2	(I)		334340029300111	H0002020093
B	DX2	(J)		334340033330011	H0001010094
B	DX2	(K)		334340035390111	H0005050095
B	DX2	(L)		334340042430111	H0002020096
B	DX2	(M)		334340046530111	H0008080097
B	DX2	(N)		334340056620111	H0007070098
B	DX2	(O)		334340065670111	H0003030099
B	DX2	(P)		334340070710111	H0002020100
B	DX2	(Q)		334340073740111	H0002020101
B	DX2	(R)		322220029300111	H0002020102
B	DX2	(S)		322220033370111	H0005050103
B	DX2	(T)		322220041440111	H0004040104
B	DX2	(U)		322220047470011	H0001010105
B	DX2	(V)		322220049490011	H0001010106

B	DX2 (W)	322220053540111	H0002020107
B	DX2 (X)	322220057610111	H0005050108
B	DX2 (Y)	322220064770111	H0014140109
B	DX2 (Z)	310100029300111	H0002020110
B	DX2 (AA)	310100032330111	H0002020111
B	DX2 (BB)	310100035360111	H0002020112
B	DX2 (CC)	310100038390111	H0002020113
B	DX2 (DD)	310100041430111	H0003030114
B	DX2 (EE)	310100045460111	H0002020115
B	DX2 (FF)	310100049490011	H0001010116
B	DX2 (GG)	310100051520111	H0002020117
B	DX2 (HH)	310100054640111	H0011110118

//EOF0503

H03830247NY0051SOCIAL WORK

*

H020445

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
B	OTHER UCA	2171205818100024H0002010005	
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
BREASON FOR #2 (1)		230300011090109	V0003000020
BREASON FOR #2 (2)		227250108080009	H0003000021
BREAS FOR 2 (COTHER)		2303000121200024V0001010022	
B	IF NOT SCHED	2171700717001024V0002010023	
BIF NOT CLINIC (1)		217170063610109	V0003000024
BIF NOT CLINIC (2)		2141201606000024H0003010025	
B	REF CODE PFX	2040400706002024V0006010026	
B	REF CODE 1	0030300716001024V0012010027	
B	REF CODE 2	0020101714501024V0054020028	
B	JOB RELATED	217170032320011	V0001010029
B	MIL ONLY DUTY	2202000241410024V0002010030	
B	MIL ONLY QTRS	2191900222001024V0003010031	
B	MIL ONLY PROF	2191900181501024V0004010032	
F	FILLEP	805	050033
B	SPEC PRE CLIN	306060022140111	H0009090034
B	NOT AVAILABLE	320200009070111	H0003030035
BPROV2 ADDL1 (YES)		208080006060011	V0001010036
B	ADDL PROC1	1110701071601024V0050050037	
BPROV2 ADDL2 (YES)		202020006060011	V0001010038
B	ADDL PROC2	1050101071601024V0050050039	
F	ADMITTED	801	010040
BUNL REAS PRIM PFX		2121200747602024V0002010041	
B	UNL REAS PRIM	1110701748301034V0050050042	
B	UNL REAS SEC PFX	2060600747602024V0002010043	
B	UNL REAS SEC	1050101748301034V0050050044	
B	FOLLOW/R OUT	2383800323101024V0002010045	
B	PROC PROV1 (A)	347470008240111	H0017170046
B	PROC PROV1 (B)	335350008080011	H0001010047
B	PROC PROV1 (C)	335350010200111	H0011110048
B	PROC PROV1 (D)	335350022220011	H0001010049
B	PROC PROV1 (E)	335350024240011	H0001010050
B	PROC PROV1 (F)	335350026280111	H0003030051

B	PROC PROV2 (A)	346460008240111	H0017170052
B	PROC PROV2 (B)	334340008080011	H0001010053
B	PROC PROV2 (C)	334340010200111	H0011110054
B	PROC PROV2 (D)	334340022220011	H0001010055
B	PROC PROV2 (E)	334340024240011	H0001010056
B	PROC PROV2 (F)	334340026280111	H0003030057
B	OTHER CODES	311110018180011	H0001010058
B	DX1 COL1 (A)	247470036380109	V0003000059
B	DX1 COL1 (B)	247470040440109	V0005000060
B	DX1 COL1 (C)	247470046470109	V0002000061
B	DX1 COL1 (D)	247470049500109	V0002000062
B	DX1 COL1 (E)	247470052530109	V0002000063
B	DX1 COL1 (F)	247470055570109	V0003000064
B	DX1 COL1 (G)	247470059600109	V0002000065
B	DX1 COL1 (H)	247470063650109	V0003000066
B	DX1 COL1 (I)	247470067670009	V0001000067
B	DX1 COL1 (J)	247470069700109	V0002000068
B	DX1 COL1 (K)	247470072750109	V0004000069
B	DX1 COL1 (L)	2474700777700024	V0001020070
B	DX1 COL2 (A)	235350034350109	V0002000071
B	DX1 COL2 (B)	235350037380109	V0002000072
B	DX1 COL2 (C)	235350040490109	V0010000073
B	DX1 COL2 (D)	235350052590109	V0008000074
B	DX1 COL2 (E)	235350062660109	V0005000075
B	DX1 COL2 (F)	235350069690009	V0001000076
B	DX1 COL2 (G)	235350071710009	V0001000077
B	DX1 COL2 (H)	235350073760109	V0004000078
B	DX1 COL2 (I)	2353500787901024	V0002020079
B	DX1 COL3 (A)	223230034350109	V0002000080
B	DX1 COL3 (B)	223230037370009	V0001000081
B	DX1 COL3 (C)	223230039390009	V0001000082
B	DX1 COL3 (D)	223230042420009	V0001000083
B	DX1 COL3 (E)	223230044460109	V0003000084
B	DX1 COL3 (F)	223230049500109	V0002000085
B	DX1 COL3 (G)	223230052520009	V0001000086
B	DX1 COL3 (H)	223230054550109	V0002000087
B	DX1 COL3 (I)	223230058640109	V0007000088
B	DX1 COL3 (J)	223230066660009	V0001000089
B	DX1 COL3 (K)	223230069700109	V0002000090
B	DX1 COL3 (L)	223230072760109	V0005000091
B	DX1 COL3 (M)	2232300787800024	V0001020092
B	DX1 COL4 (A)	211110034350109	V0002000093
B	DX1 COL4 (B)	211110037370009	V0001000094
B	DX1 COL4 (C)	211110039390009	V0001000095
B	DX1 COL4 (D)	211110042420009	V0001000096
B	DX1 COL4 (E)	211110044450109	V0002000097
B	DX1 COL4 (F)	211110047480109	V0002000098
B	DX1 COL4 (G)	211110050500009	V0001000099
B	DX1 COL4 (H)	211110053550109	V0003000100
B	DX1 COL4 (I)	211110058600109	V0003000101
B	DX1 COL4 (J)	211110062640109	V0003000102
B	DX1 COL4 (K)	2111100666801024	V0003020103
B	DX2 (A)	346460036380111	H0003030104
B	DX2 (B)	346460040440111	H0005050105
B	DX2 (C)	346460046470111	H0002020106

B	DX2 (D)	346460049500111	H0002020107
B	DX2 (E)	346460052530111	H0002020108
B	DX2 (F)	346460055570111	H0003030109
B	DX2 (G)	346460059600111	H0002020110
B	DX2 (H)	346460063650111	H0003030111
B	DX2 (I)	346460067670011	H0001010112
B	DX2 (J)	346460069700111	H0002020113
B	DX2 (K)	346460072750111	H0004040114
B	DX2 (L)	346460077770011	H0001010115
B	DX2 (M)	334340034350111	H0002020116
B	DX2 (N)	334340037380111	H0002020117
B	DX2 (O)	334340040490111	H0010100118
B	DX2 (P)	334340052590111	H0008080119
B	DX2 (Q)	334340062660111	H0005050120
B	DX2 (R)	334340069690011	H0001010121
B	DX2 (S)	334340071710011	H0001010122
B	DX2 (T)	334340073760111	H0004040123
B	DX2 (U)	334340078790111	H0002020124
B	DX2 (V)	322220034350111	H0002020125
B	DX2 (W)	322220037370011	H0001010126
B	DX2 (X)	322220039390011	H0001010127
B	DX2 (Y)	322220042420011	H0001010128
B	DX2 (Z)	322220044460111	H0003030129
B	DX2 (AA)	322220049500111	H0002020130
B	DX2 (BB)	322220052520011	H0001010131
B	DX2 (CC)	322220054550111	H0002020132
B	DX2 (DD)	322220058640111	H0007070133
B	DX2 (EE)	322220066660011	H0001010134
B	DX2 (FF)	322220069700111	H0002020135
B	DX2 (GG)	322220072760111	H0005050136
B	DX2 (HH)	322220078780011	H0001010137
B	DX2 (II)	310100034350111	H0002020138
B	DX2 (JJ)	310100037370011	H0001010139
B	DX2 (KK)	310100039390011	H0001010140
B	DX2 (LL)	310100042420011	H0001010141
B	DX2 (MM)	310100044450111	H0002020142
B	DX2 (NN)	310100047480111	H0002020143
B	DX2 (OO)	310100050500011	H0001010144
B	DX2 (PP)	310100053550111	H0003030145
B	DX2 (QQ)	310100058600111	H0003030146
B	DX2 (RR)	310100062640111	H0003030147
B	DX2 (SS)	310100066670111	H0002020148
B	SPEC PROG (1)	307070035320111	H0004040149
B	SPEC PROG (2)	304020131310011	V0003030150

//EOF0529

H03830247NY0051UROLOGY

*

H010231

*

B	LITHO	739160104040011	H0024080001
B	UCA PFX	2090900706202024V0005010002	
B	UCA CODE 1	0080800716001024V0012010003	
B	UCA CODE 2	0070601714501024V0054020004	
F	OTHER UCA	801	010005
B	DATE (MO)	2434300574601014V0012020006	
B	DATE (DY)1	1454500555201014V0004010007	
B	DATE (DY)2	1444400554601014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501554601014V0090090012	
B	FMP	1373601554601014V0020020013	
B	VISIT COUNT	2141400544601024V0009010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
	BREASON FOR #2 (1)	230300011090109	V0003000020
	BREASON FOR #2 (2)	2272501080800024H0003010021	
B	IF NOT SCHED	2171700717001024V0002010022	
B	BIF NOT CLINIC (1)	217170063610109	V0003000023
B	BIF NOT CLINIC (2)	2141201606000024H0003010024	
B	REF CODE PFX	2040400706002024V0006010025	
B	REF CODE 1	0030300716001024V0012010026	
B	REF CODE 2	0020101714501024V0054020027	
B	JOB RELATED	217170032320011	V0001010028
B	MIL ONLY DUTY	2202000241410024V0002010029	
B	MIL ONLY QTRS	2191900222001024V0003010030	
B	MIL ONLY PROF	2191900181501024V0004010031	
F	FILLER	805	050032
B	SPEC PRE CLIN	306060022140111	H0009090033
B	NOT AVAILABLE	320200009070111	H0003030034
B	BPROV2 ADDL1 (YES)	208080006060011	V0001010035
B	ADDL PROC1	1110701071601024V0050050036	
B	BPROV2 ADDL2 (YES)	202020006060011	V0001010037
B	ADDL PROC2	1050101071601024V0050050038	
B	ADMITTED	211110019190011	V0001010039
B	BUNL REAS PRIM PFX	2121200747602024V0002010040	
B	UNL REAS PRIM	1110701748301034V0050050041	
B	UNL REAS SEC PFX	2060600747602024V0002010042	
B	UNL REAS SEC	1050101748301034V0050050043	
B	FOLLOW/R OUT	2383800393801024V0002010044	
B	PROC PROV1 (A)	347470008090111	H0002020045
B	PROC PROV1 (B)	347470011200111	H0010100046
B	PROC PROV1 (C)	347470023230011	H0001010047
B	PROC PROV1 (D)	347470025300111	H0006060048
B	PROC PROV1 (E)	347470032320011	H0001010049
B	PROC PROV1 (F)	347470034350111	H0002020050
B	PROC PROV1 (G)	335350008120111	H0005050051

B	PROC PROV1	(H)	335350014150111	H0002020052
B	PROC PROV1	(I)	335350017290111	H0013130053
B	PROC PROV1	(J)	323230009100111	H0002020054
B	PROC PROV1	(K)	323230012150111	H0004040055
B	PROC PROV1	(L)	323230017220111	H0006060056
B	PROC PROV1	(M)	323230024250111	H0002020057
B	PROC PROV2	(A)	346460008090111	H0002020058
B	PROC PROV2	(B)	346460011200111	H0010100059
B	PROC PROV2	(C)	346460023230011	H0001010060
B	PROC PROV2	(D)	346460025300111	H0006060061
B	PROC PROV2	(E)	346460032320011	H0001010062
B	PROC PROV2	(F)	346460034350111	H0002020063
B	PROC PROV2	(G)	334340008120111	H0005050064
B	PROC PROV2	(H)	334340014150111	H0002020065
B	PROC PROV2	(I)	334340017290111	H0013130066
B	PROC PROV2	(J)	322220009100111	H0002020067
B	PROC PROV2	(K)	322220012150111	H0004040068
B	PROC PROV2	(L)	322220017220111	H0006060069
B	PROC PROV2	(M)	322220024250111	H0002020070
BDX1	COL1	(A)	247470043490109	V0007000071
BDX1	COL1	(B)	247470051520109	V0002000072
BDX1	COL1	(C)	247470054570109	V0004000073
BDX1	COL1	(D)	2474700606501024	V0006020074
BDX1	COL2	(A)	235350041420109	V0002000075
BDX1	COL2	(B)	235350044490109	V0006000076
BDX1	COL2	(C)	235350052600109	V0009000077
BDX1	COL2	(D)	235350062630109	V0002000078
BDX1	COL2	(E)	2353500666901024	V0004020079
BDX1	COL3	(A)	223230041530109	V0013000080
BDX1	COL3	(B)	223230055580109	V0004000081
BDX1	COL3	(C)	223230060610109	V0002000082
BDX1	COL3	(D)	223230063640109	V0002000083
BDX1	COL3	(E)	223230066690109	V0004000084
BDX1	COL3	(F)	223230071750109	V0005000085
BDX1	COL3	(G)	223230077770009	V0001000086
BDX1	COL3	(H)	2232300798001024	V0002020087
BDX1	COL4	(A)	211110042510109	V0010000088
BDX1	COL4	(B)	211110053530009	V0001000089
BDX1	COL4	(C)	2111100555500024	V0001020090
B	DX2	(A)	346460043490111	H0007070091
B	DX2	(B)	346460051520111	H0002020092
B	DX2	(C)	346460054570111	H0004040093
B	DX2	(D)	346460060650111	H0006060094
B	DX2	(E)	334340041420111	H0002020095
B	DX2	(F)	334340044490111	H0006060096
B	DX2	(G)	334340052600111	H0009090097
B	DX2	(H)	334340062630111	H0002020098
B	DX2	(I)	334340066690111	H0004040099
B	DX2	(J)	322220041530111	H0013130100
B	DX2	(K)	322220055580111	H0004040101
B	DX2	(L)	322220060610111	H0002020102
B	DX2	(M)	322220063640111	H0002020103
B	DX2	(N)	322220066690111	H0004040104
B	DX2	(O)	322220071750111	H0005050105
B	DX2	(P)	322220077770011	H0001010106

B	DX2 (Q)	322220079800111	H0002020107
B	DX2 (R)	310100042510111	H0010100108
B	DX2 (S)	310100053530011	H0001010109

//EOF0543

H02830247NY0051BAS/TMC

*

H0546

*

B LITHO 739160104040011 H0024080001
B DATE (MO) 2070700716001014V0012020002
B DATE (DY) 1090801696001014V0020020003
B SSN 1090101544501014V0090090004
B APPT. STATUS 2121200413901024V0003010005
B INJURIES (A) 206060042380109 V0005000006
B INJURIES (B) 2060600363600024V0001010007
B JOB RELATED 206060029290011 V0001010008
BMILITARY DUTY (A) 209090024141009 V0002000009
BMILITARY DUTY (B) 2090900131300024V0001010010
BMILITARY QUARTERS 2080800222001024V0003010011
B MILITARY PROFILE 2080800181501024V0004010012
BTMC DISPOSITION 209090025250011 V0001010013
BPREASSIGNED CODES 301010024160111 H0009090014
B BAS UCA 0434201785201024V0054020015
BBAS PROV1 PFX (A) 037370077640109 V0014000016
BBAS PROV1 PFX (B) 0363600766401024V0013010017
B BAS PROV1 CODE 1333001776801024V0040040018
B BAS PROV1 TIME 2272700797101024V0009010019
BBAS PROV2 PFX (A) 021210077640109 V0014000020
BBAS PROV2 PFX (B) 0202000766401024V0013010021
B BAS PROV2 CODE 1171401776801024V0040040022
B BAS PROV2 TIME 2242400797101024V0009010023
BBAS REAS FOR #2 A 228280067650109 V0003000024
BBAS REAS FOR #2 B 2262401646400024H0003010025
BBASPROV2 ADDLPROC12242400616100024V0001010026
B BAS ADDL PROC 1 1282401584901024V0050050027
BBASPROV2 ADDLPROC22171700616100024V0001010028
B BAS ADDL PROC 2 1211701584901024V0050050029
B BAS FIELD 243430050500011 V0001010030
B TMC UCA PFX 2444400403901024V0002010031
B TMC UCA CODE (A) 0434300413901024V0003010032
B TMC UCA CODE (B) 0424200411501024V0027010033
BTMC PROV1 PFX (A) 037370041280109 V0014000034
BTMC PROV1 PFX (B) 0363600402801024V0013010035
B TMC PROV1 CODE 1333001413201024V0040040036
B TMC PROV1 TIME 2272700433501024V0009010037
BTMC PROV2 PFX (A) 021210041280109 V0014000038
BTMC PROV2 PFX (B) 0202000402801024V0013010039
B TMC PROV2 CODE 1171401413201024V0040040040
B TMC PROV2 TIME 2242400433501024V0009010041
BTMC REAS FOR #2 A 228280031290109 V0003000042
BTMC REAS FOR #2 B 2262401282800024H0003010043
BTMC PROV2 ADDL PRO212120023230011 V0001010044
B TMC ADDL PROC 1161201201101024V0050050045
B TMC ADMITTED 228280007070011 V0001010046
BTMC PROC PROV1 (A)340400024150111 H0010100047
BTMC PROC PROV1 (B)340400011070111 H0005050048
BTMC PROC PROV1 (C)328280024150111 H0010100049
BTMC PROC PROV2 (A)339390024150111 H0010100050
BTMC PROC PROV2 (B)339390011070111 H0005050051

BTMC PROC PROV2 (C)	327270024150111	H0010100052
B FOLLOW/R OUT	2424200050401024V0002010053	
B PRIM UNL REAS PFX	2121200747602024V0002010054	
B PRIM UNL REAS	1110701748301034V0050050055	
B SEC UNL REAS PFX	2060600747602022V0002010056	
B SEC UNL REAS	1050101748301034V0050050057	
B DX1 COL1 (A)	247470009120109	V0004000058
B DX1 COL1 (B)	247470016220109	V0007000059
B DX1 COL1 (C)	247470024250109	V0002000060
B DX1 COL1 (D)	247470027310109	V0005000061
B DX1 COL1 (E)	247470033340109	V0002000062
B DX1 COL1 (F)	247470036410109	V0006000063
B DX1 COL1 (G)	247470043460109	V0004000064
B DX1 COL1 (H)	2474700495501024V0007020065	
B DX1 COL2 (A)	235350007140109	V0008000066
B DX1 COL2 (B)	235350018220109	V0005000067
B DX1 COL2 (C)	235350024260109	V0003000068
B DX1 COL2 (D)	2353500294201024V0014020069	
B DX1 COL3 (A)	223230007250109	V0019000070
B DX1 COL3 (B)	223230028340109	V0007000071
B DX1 COL3 (C)	223230037490109	V0013000072
B DX1 COL3 (D)	2232300525401024V0003020073	
B DX1 COL4 (A)	211110007110109	V0005000074
B DX1 COL4 (B)	211110013180109	V0006000075
B DX1 COL4 (C)	211110022220009	V0001000076
B DX1 COL4 (D)	211110024240009	V0001000077
B DX1 COL4 (E)	211110027280109	V0002000078
B DX1 COL4 (F)	211110030350109	V0006000079
B DX1 COL4 (G)	211110037390109	V0003000080
B DX1 COL4 (H)	211110042470109	V0006000081
B DX1 COL4 (I)	211110050510109	V0002000082
B DX1 COL4 (J)	211110053580109	V0006000083
B DX1 COL4 (K)	2111100606101024V0002020084	
B DX2 (A)	346460009120111	H0004040085
B DX2 (B)	346460016220111	H0007070086
B DX2 (C)	346460024250111	H0002020087
B DX2 (D)	346460027310111	H0005050088
B DX2 (E)	346460033340111	H0002020089
B DX2 (F)	346460036410111	H0006060090
B DX2 (G)	346460043460111	H0004040091
B DX2 (H)	346460049550111	H0007070092
B DX2 (I)	334340007140111	H0008080093
B DX2 (J)	334340018220111	H0005050094
B DX2 (K)	334340024260111	H0003030095
B DX2 (L)	334340029420111	H0014140096
B DX2 (M)	322220007250111	H0019190097
B DX2 (N)	322220028340111	H0007070098
B DX2 (O)	322220037490111	H0013130099
B DX2 (P)	322220052540111	H0003030100
B DX2 (Q)	310100007110111	H0005050101
B DX2 (R)	310100013180111	H0006060102
B DX2 (S)	310100022220011	H0001010103
B DX2 (T)	310100024240011	H0001010104
B DX2 (U)	310100027280111	H0002020105
B DX2 (V)	310100030350111	H0006060106

B	DX2 (W)	310100037390111	H0003030107
B	DX2 (X)	310100042470111	H0006060108
B	DX2 (Y)	310100050510111	H0002020109
B	DX2 (Z)	310100053580111	H0006060110
B	NOT AVAILABLE	312120035330111	H0003030111

//EOF0308

820 EKG

SCAN820.FIL

H02830247NY0051EKG

*

H0541

*

B LITHO 739160104040011 H0024080001
B OTHER UCA 1 208080081770209 V0003000002
B OTHER UCA 2 2040400817702024V0003010003
B DATE (MO) 2424200817001014V0012020004
B DATE (DY) 1464402797001014V0020020005
B PROV1 PFX (1) 035350079660109 V0014000006
B PROV1 PFX (2) 0343400786601014V0013010007
B PROV1 CODE 1322602786901014V0040040008
B SSN (1) 1422602605101014V0090090009
B FMP (1) 1464402605101014V0020020010
BREF CODE SINGLE 1A232320047370209 V0006000011
BREF CODE SINGLE 1B2282800473702024V0006020012
BREFCODE HAND PFX 12323200322402024V0005010013
B REF CODE HAND 1 0302602330701024V0081030014
B TIME SPENT 1A 246460014120109 V0003000015
B TIME SPENT 1B 242420014120109 V0003000016
B TIME SPENT 1C 2383800141201024V0003010017
B UNLISTED PROC 1 1463802261701024V0050050018
B PROCEDURES 1 346460048310111 H0018180019
B SP PRE CLIN 1 345370107070011 V0009090020
B SSN 2 1180202605101024V0090090021
B FMP 2 1222002605101024V0020020022
BREF CODE SINGLE 2A208080047370209 V0006000023
BREF CODE SINGLE 2B2040400473702024V0006020024
BREFCODE HAND PFX 22080800322402024V0005010025
B REF CODE HAND 2 0060202330701024V0081030026
B TIME 2A 222220014120109 V0003000027
B TIME 2B 218180014120109 V0003000028
B TIME 2C 2141400141201024V0003010029
B UNLISTED PROC 2 1221402261701024V0050050030
B PROCEDURES 2 322220048310111 H0018180031
B SP PRE CLIN 2 321130107070011 V0009090032
B SSN 3 1432702152401024V0090090033
B FMP 3 1474502152401024V0020020034
BREFCODE SINGLE 3A 233330028380209 V0006000035
BREFCODE SINGLE 3B 2292900283802024V0006020036
BREFCODE HAND PFX 32333300435102024V0005010037
B REF CODE HAND 3 0312702426801024V0081030038
B TIME SPENT 3A 247470061630109 V0003000039
B TIME SPENT 3B 243430061630109 V0003000040
B TIME SPENT 3C 2393900616301024V0003010041
B UNLISTED PROC 3 1473902495801024V0050050042
B PROCEDURES 3 347470027440111 H0018180043
B SP PRE CLIN 3 346380168680011 V0009090044
B SSN 4 1190302152401024V0090090045
B FMP 4 1232102152401024V0020020046
BREF CODE SINGLE 4A209090028380209 V0006000047
BREF CODE SINGLE 4B2050500283802024V0006020048
BREFCODE HAND PFX 42090900435102024V0005010049
B REF CODE HAND 4 0070302426801024V0081030050
B TIME SPENT 4A 223230061630109 V0003000051

B TIME SPENT 4B 219190061630109 V0003000052
B TIME SPENT 4C 2151500616301024V0003010053
B UNLISTED PROC 4 1231502495801024V0050050054
B PROCEDURES 4 323230027440111 H0018180055
B SP PRE CLIN 4 322140168680011 V0009090056
//EOF0218

830 EMERGENCY ROOM

SCAN830.FIL

H02830247NY0051EMERGENCY ROOM

*

H0540

*

B	LITHO	739160104040011	H0024080001
F	CLINIC PFX	801	010002
F	CLINIC CODE	803	030003
F	OTHER UCA	801	010004
B	DATE (MO)	2191900705901024	V0012020005
B	DATE (DY)1	1212100686501014	V0004010006
B	DATE (DY)2	1202000685901014	V0010010007
B	PROV1 PFX1	045450077640109	V0014000008
B	PROV1 PFX2	0444400766401014	V0013010009
B	PROV1 CODE	1434001776801014	V0040040010
B	SSN	1090101635401014	V0090090011
B	FMP	1131201635401014	V0020020012
F	VISIT COUNT	801	010013
B	PROV1 TIME	2383800795801014	V0022020014
B	PROV2 PFX1	030300077640109	V0014000015
B	PROV2 PFX2	0292900766401024	V0013010016
B	PROV2 CODE	1282501776801024	V0040040017
B	PROV2 TIME	2323200795801024	V0022020018
B	BREASON FOR #2 (A)	231310055530109	V0003000019
B	BREASON FOR #2 (B)	2282601525200024	H0003010020
B	IF NOT EMERGENCY	205050046460011	V0001010021
B	SEEN IN HOSPITAL	217170046460011	V0001010022
F	REF CODE PFX	801	010023
F	REF CODE	803	030024
B	JOB RELATED	229290039390011	V0001010025
B	MIL ONLY DUTY	2444400483810024	V0002010026
B	MIL ONLY QTRS	2434300464401024	V0003010027
B	MIL ONLY PROF	2434300423901024	V0004010028
F	NOT AVAIL	801	010029
B	SP PRE CLIN	309010138380011	V0009090030
F	SPEC PROG	807	070031
B	PROV2 ADDL1	208080016160011	V0001010032
B	ADDL PROC1	1110701150601024	V0050050033
B	PROV2 ADDL2	202020016160011	V0001010034
B	ADDL PROC2	1050101150601024	V0050050035
B	ADMITTED	209090029290011	V0001010036
B	BUNL REAS PRIM PFX	2121200747602024	V0002010037
B	UNL REAS PRIM	1110701748301034	V0050050038
B	UNL REAS SEC PFX	2060600747602024	V0002010039
B	UNL REAS SEC	1050101748301034	V0050050040
B	FOLLOW/R OUT	2464600050401024	V0002010041
B	PROC PROV1 (A)	345450030080111	H0023230042
B	PROC PROV1 (B)	333330029230111	H0007070043
B	PROC PROV1 (C)	333330021060111	H0016160044
B	PROC PROV1 (D)	321210030060111	H0025250045
B	PROC PROV2 (A)	344440030080111	H0023230046
B	PROC PROV2 (B)	332320029230111	H0007070047
B	PROC PROV2 (C)	332320021060111	H0016160048
B	PROC PROV2 (D)	320200030060111	H0025250049
B	DX1 COL1 (A)	247470009130109	V0005000050
B	DX1 COL1 (B)	247470017240109	V0008000051

B	DX1 COL1 (C)	247470026320109	V0007000052
B	DX1 COL1 (D)	247470034390109	V0006000053
B	DX1 COL1 (E)	247470041410009	V0001000054
B	DX1 COL1 (F)	247470043460109	V0004000055
B	DX1 COL1 (G)	247470049570109	V0009000056
B	DX1 COL1 (H)	2474700607401024	V0015020057
B	DX1 COL2 (A)	235350008140109	V0007000058
B	DX1 COL2 (B)	235350016210109	V0006000059
B	DX1 COL2 (C)	235350024320109	V0009000060
B	DX1 COL2 (D)	235350035390109	V0005000061
B	DX1 COL2 (E)	235350042550109	V0014000062
B	DX1 COL2 (F)	235350058690109	V0012000063
B	DX1 COL2 (G)	2353500728101024	V0010020064
B	DX1 COL3 (A)	223230007080109	V0002000065
B	DX1 COL3 (B)	223230012140109	V0003000066
B	DX1 COL3 (C)	223230016180109	V0003000067
B	DX1 COL3 (D)	223230021370109	V0017000068
B	DX1 COL3 (E)	223230039400109	V0002000069
B	DX1 COL3 (F)	223230042470109	V0006000070
B	DX1 COL3 (G)	223230049520109	V0004000071
B	DX1 COL3 (H)	223230055580109	V0004000072
B	DX1 COL3 (I)	223230061660109	V0006000073
B	DX1 COL3 (J)	2232300687001024	V0003020074
B	DX1 COL4 (A)	211110021240109	V0004000075
B	DX1 COL4 (B)	211110026260009	V0001000076
B	DX1 COL4 (C)	211110028280009	V0001000077
B	DX1 COL4 (D)	211110030300009	V0001000078
B	DX1 COL4 (E)	211110032360109	V0005000079
B	DX1 COL4 (F)	211110038380009	V0001000080
B	DX1 COL4 (G)	211110040450109	V0006000081
B	DX1 COL4 (H)	211110047490109	V0003000082
B	DX1 COL4 (I)	211110052540109	V0003000083
B	DX1 COL4 (J)	2111100565600024	V0001020084
B	DX2 (A)	346460009130111	H0005050085
B	DX2 (B)	346460017240111	H0008080086
B	DX2 (C)	346460026320111	H0007070087
B	DX2 (D)	346460034390111	H0006060088
B	DX2 (E)	346460041410011	H0001010089
B	DX2 (F)	346460043460111	H0004040090
B	DX2 (G)	346460049570111	H0009090091
B	DX2 (H)	346460060740111	H0015150092
B	DX2 (I)	334340008140111	H0007070093
B	DX2 (J)	334340016210111	H0006060094
B	DX2 (K)	334340024320111	H0009090095
B	DX2 (L)	334340035390111	H0005050096
B	DX2 (M)	334340042550111	H0014140097
B	DX2 (N)	334340058690111	H0012120098
B	DX2 (O)	334340072810111	H0010100099
B	DX2 (P)	322220007080111	H0002020100
B	DX2 (Q)	322220012140111	H0003030101
B	DX2 (R)	322220016180111	H0003030102
B	DX2 (S)	322220021370111	H0017170103
B	DX2 (T)	322220039400111	H0002020104
B	DX2 (U)	322220042470111	H0006060105
B	DX2 (V)	322220049520111	H0004040106

B	DX2 (W)	322220055580111	H0004040107
B	DX2 (X)	322220061660111	H0006060108
B	DX2 (Y)	322220068700111	H0003030109
B	DX2 (Z)	310100021240111	H0004040110
B	DX2 (AA)	310100026260011	H0001010111
B	DX2 (BB)	310100028280011	H0001010112
B	DX2 (CC)	310100030300011	H0001010113
B	DX2 (DD)	310100032360111	H0005050114
B	DX2 (EE)	310100038380011	H0001010115
B	DX2 (FF)	310100040450111	H0006060116
B	DX2 (GG)	310100047490111	H0003030117
B	DX2 (HH)	310100052540111	H0003030118

//EOF0440

840 GROUP FORM I

SCAN840.FIL

H02830247NY0051GROUP FORM I

*

H0146

*

B	LITHO	746210104040011	H0026080001
B	UCA PFX	2080800787002014	V0005010002
B	UCA CODE	0060202795301014	V00081030003
B	DATE (MO)	2292900776601014	V0012020004
B	DATE (DY1)	1313100757201014	V0004010005
B	DATE (DY2)	1303000756601014	V0010010006
B	PROV1 PFX1	047470079660109	V0014000007
B	PROV1 PFX2	0464600786601014	V0013010008
B	PROV1 CODE	1454201797001014	V0040040009
B	VISIT COUNT	2151500777106024	V0002010010
B	PROV1 TIME SPENT	1464401574801014	V0030030011
B	PROV2 PFX1	040400079660109	V0014000012
B	PROV2 PFX2	0393900786601024	V0013010013
B	PROV2 CODE	1383501797001024	V0040040014
B	PROV2 TIME SPENT	1383601574801024	V0030030015
B	UNIT ID CODE	0120202450901024	V0222060016
B	TIME PREP (1)	1464401372801024	V0030030017
B	TIME PREP (2)	1383601372801024	V0030030018
B	TIME TRAV (1)	1464401170801024	V0030030019
B	TIME TRAV (2)	1383601170801024	V0030030020
B#	ARMY ACTIVE DUT	1282601524301024	V0030030021
B#	OTHER ACTIVE DUT	1282601342501024	V0030030022
B	# RETIRED MILT	1222001342501024	V0030030023
B	# DEPENDENTS	1282601170801024	V0030030024
B	# CIVILIANS	1222001170801024	V0030030025
B	CONTIN SHEET USED	211110004040011	V0001010026
B	UNL REASON (PROC)	1090102091801024	V0050050027
B	REASON (PROC) A	247470006100109	V0005000028
B	REASON (PROC) B	247470013170109	V0005000029
B	REASON (PROC) C	247470019230109	V0005000030
B	REASON (PROC) D	247470025350109	V0011000031
B	REASON (PROC) E	247470037370009	V0001000032
B	REASON (PROC) F	247470039400109	V0002000033
B	REASON (PROC) G	247470042450109	V0004000034
B	REASON (PROC) H	247470047510109	V0005000035
B	REASON (PROC) I	235350006110109	V0006000036
B	REASON (PROC) J	235350014190109	V0006000037
B	REASON (PROC) K	235350022250109	V0004000038
B	REASON (PROC) L	235350027300109	V0004000039
B	REASON (PROC) M	235350032330109	V0002000040
B	REASON (PROC) N	235350035370109	V0003000041
B	REASON (PROC) O	235350039390009	V0001000042
B	REASON (PROC) P	235350042440109	V0003000043
B	REASON (PROC) Q	235350047500109	V0004000044
B	REASON (PROC) R	235350052530109	V0002000045
B	REASON (PROC) S	223230006150109	V0010000046
B	REASON (PROC) T	223230018240109	V0007000047
B	REASON (PROC) U	223230027320109	V0006000048
B	REASON (PROC) V	223230034340009	V0001000049
B	REASON (PROC) W	2232300383800024	V0001010050
F	FILLER	849	490051

F FILLER 849
F FILLER 801
//EOF0172

490052
010053

860 IMMUNIZATION

SCAN860.FIL

H02830247NY0051IMMUNIZATION

*

H0544

*

B	LITHO	739160104040011	H0024080001
B	CLINIC PREFIX	2080800776902024	V0005010002
B	CLINIC CODE	0060202785201024	V0081030003
F	OTHER UCA	801	010004
B	DATE (MO)	2383800635201014	V0012020005
B	DATE (DY)	1424002615201014	V0020020006
B	PROV1 PFX	0464600785201014	V0027010007
B	PROV1 CODE	1443802786901014	V0040040008
B	SSN A	1422602382901014	V0090090009
B	FMP A	1464402382901014	V0020020010
B	PROC A1	324240046290111	H0018180011
B	PROC A2	314140047310111	H0017170012
B	ALLERGY A	310070129290011	V0004040013
B	SPE PRE CODE A	302020037290111	H0009090014
B	SSN B	1422602170801024	V0090090015
B	FMP B	1464402170801024	V0020020016
B	PROC B1	324240025080111	H0018180017
B	PROC B2	314140026100111	H0017170018
B	ALLERGY B	310070108080011	V0004040019
B	SPE PRE CODE B	302020016080111	H0009090020
B	SSN C	1432702142301024	V0090090021
B	FMP C	1474502142301024	V0020020022
B	PROC C1	325250006230111	H0018180023
B	PROC C2	315150005210111	H0017170024
B	ALLERGY C	311080123230011	V0004040025
B	SPE PRE CODE C	303030015230111	H0009090026
B	SSN D	1432702344301024	V0090090027
B	FMP D	1474502344301024	V0020020028
B	PROC D1	325250026430111	H0018180029
B	PROC D2	315150025410111	H0017170030
B	ALLERGY D	311080143430011	V0004040031
B	SPE PRE CODE D	303030035430111	H0009090032
B	SSN E	1432702546301024	V0090090033
B	FMP E	1474502546301024	V0020020034
B	PROC E1	325250046630111	H0018180035
B	PROC E2	315150045610111	H0017170036
B	ALLERGY E	311080163630011	V0004040037
B	SPE PRE CODE E	303030055630111	H0009090038
B	SSN F	1432702748301024	V0090090039
B	FMP F	1474502748301024	V0020020040
B	PROC F1	325250066830111	H0018180041
B	PROC F	315150065810111	H0017170042
B	ALLERGY F	311080183830011	V0004040043
B	SPE PRE CODE F	303030075830111	H0009090044

//EOF0376

900 OCCUPATIONAL HEALTH

SCAN900.FIL

H02830247NY0051OCCUPATIONAL HEALTH

*

H0538

*

B	LITHO	739160104040011	H0024080001
B	CLINIC PFX	2040400706202024V0005010002	
B	CLINIC CODE	0030101714501024V0081030003	
B	OTHER UCA 1	212120081800109	V0002000004
B	OTHER UCA 2	2060600818100024V0001010005	
B	DATE (MO)	2434300604901014V0012020006	
B	DATE (DY) 1	1454500585501014V0004010007	
B	DATE (DY) 2	1444400584901014V0010010008	
B	PROV1 PFX1	044440033200109	V0014000009
B	PROV1 PFX2	0434300322001014V0013010010	
B	PROV1 CODE	1423901332401014V0040040011	
B	SSN	1332501584901014V0090090012	
B	FMP	1373601584901014V0020020013	
B	JOB REL IF YES	2202000323001024V0003010014	
B	PROV1 TIME	2373700351401014V0022020015	
B	PROV2 PFX1	029290033200109	V0014000016
B	PROV2 PFX2	0282800322001024V0013010017	
B	PROV2 CODE	1272401332401024V0040040018	
B	PROV2 TIME	2313100351401024V0022020019	
B	REAS FOR #2 A	230300011090109	V0003000020
B	REAS FOR #2 B	2272501080800024H0003010021	
B	IF NOT SCHED	2121200717001024V0002010022	
B	IF NOT CLINIC 1	212120063610109	V0003000023
B	IF NOT CLINIC 2	2090701606000024H0003010024	
B	PURPOSE ILL	2161204555500024H0002010025	
B	PURPOSE INJ	2161204545400024H0002010026	
B	PURPOSE A	020200054530109	V0002000027
B	PURPOSE B	020200051490109	V0003000028
B	PURPOSE C	020200047460109	V0002000029
B	PURPOSE D	0131300534701024V0007010030	
B	JOB REL	217170034340011	V0001010031
B	MIL ONLY DUTY	2202000241410024V0002010032	
B	MIL ONLY QTRS	2191900222001024V0003010033	
B	MIL ONLY PROF	2191900181501024V0004010034	
F	FILLER	805	050035
B	SPEC PRE CLIN	344360108080011	V0009090036
B	NOT AVAIL	320200009070111	H0003030037
B	PROV1 ADDL1 OUT	2110704060600024H0002010038	
B	ADDL PROC1	1110701071601024V0050050039	
B	PROV2 ADDL2 OUT	2050104060600024H0002010040	
B	ADDL PROC2	1050101071601024V0050050041	
F	ADMITTED	801	010042
B	BUNL REAS PRIM PFX	2121200747602024V0002010043	
B	UNL REAS PRIM	1110701748301034V0050050044	
B	UNL REAS SEC PFX	2060600747602024V0002010045	
B	UNL REAS SEC	1050101748301034V0050050046	
B	R/O	238380036360011	V0001010047
B	PROC PROV1 (A)	347470009170111	H0009090048
B	PROC PROV1 (B)	347470019250111	H0007070049
B	PROC PROV1 (C)	347470027320111	H0006060050
B	PROC PROV1 (D)	335350008100111	H0003030051

B	PROC PROV1 (E)	335350012230111	H0012120052
B	PROC PROV1 (F)	335350025320111	H0008080053
B	PROC PROV1 (G)	323230008110111	H0004040054
B	PROC PROV1 (H)	323230013200111	H0008080055
B	PROC PROV1 (I)	323230022320111	H0011110056
B	PROC PROV1 (J)	311110019310111	H0013130057
B	PROC PROV2 (A)	346460009170111	H0009090058
B	PROC PROV2 (B)	346460019250111	H0007070059
B	PROC PROV2 (C)	346460027320111	H0006060060
B	PROC PROV2 (D)	334340008100111	H0003030061
B	PROC PROV2 (E)	334340012230111	H0012120062
B	PROC PROV2 (F)	334340025320111	H0008080063
B	PROC PROV2 (G)	322220008110111	H0004040064
B	PROC PROV2 (H)	322220013200111	H0008080065
B	PROC PROV2 (I)	322220022320111	H0011110066
B	PROC PROV2 (J)	310100019310111	H0013130067
B	PROC OUT (A)	345450009170111	H0009090068
B	PROC OUT (B)	345450019250111	H0007070069
B	PROC OUT (C)	345450027320111	H0006060070
B	PROC OUT (D)	333330008100111	H0003030071
B	PROC OUT (E)	333330012230111	H0012120072
B	PROC OUT (F)	333330025320111	H0008080073
B	PROC OUT (G)	321210008110111	H0004040074
B	PROC OUT (H)	321210013200111	H0008080075
B	PROC OUT (I)	321210022320111	H0011110076
B	PROC OUT (J)	309090019310111	H0013130077
B	OTHER CODES (A)	343430042390111	H0004040078
B	OTHER CODES (B)	337370042390111	H0004040079
B	OTHER CODES (C)	331310042390111	H0004040080
B	OTHER CODES (D)	325250042390111	H0004040081
B	OTHER CODES (E)	319190042390111	H0004040082
B	DX1 COL1 (A)	247470040430109	V0004000083
B	DX1 COL1 (B)	247470046480109	V0003000084
B	DX1 COL1 (C)	247470051570109	V0007000085
B	DX1 COL1 (D)	247470061630109	V0003000086
B	DX1 COL1 (E)	247470066730109	V0008000087
B	DX1 COL1 (F)	2474700768301024	V0008020088
B	DX1 COL2 (A)	235350038430109	V0006000089
B	DX1 COL2 (B)	235350046560109	V0011000090
B	DX1 COL2 (C)	235350058610109	V0004000091
B	DX1 COL2 (D)	235350064660109	V0003000092
B	DX1 COL2 (E)	235350069710109	V0003000093
B	DX1 COL2 (F)	2353500748201024	V0009020094
B	DX1 COL3 (A)	223230038510109	V0014000095
B	DX1 COL3 (B)	223230054710109	V0018000096
B	DX1 COL3 (C)	223230073730009	V0001000097
B	DX1 COL3 (D)	223230076790109	V0004000098
B	DX1 COL3 (E)	2232300818201024	V0002020099
B	DX1 COL4 (A)	211110037470109	V0011000100
B	DX1 COL4 (B)	211110049490009	V0001000101
B	DX1 COL4 (C)	211110052550109	V0004000102
B	DX1 COL4 (D)	211110057610109	V0005000103
B	DX1 COL4 (E)	2111100646801024	V0005020104
B	DX2 (A)	346460040430111	H0004040105
B	DX2 (B)	346460046480111	H0003030106

B	DX2 (C)	346460051570111	H0007070107
B	DX2 (D)	346460061630111	H0003030108
B	DX2 (E)	346460066730111	H0008080109
B	DX2 (F)	346460076830111	H0008080110
B	DX2 (G)	334340038430111	H0006060111
B	DX2 (H)	334340046560111	H0011110112
B	DX2 (I)	334340058610111	H0004040113
B	DX2 (J)	334340064660111	H0003030114
B	DX2 (K)	334340069710111	H0003030115
B	DX2 (L)	334340074820111	H0009090116
B	DX2 (M)	322220038510111	H0014140117
B	DX2 (N)	322220054710111	H0018180118
B	DX2 (O)	322220073730011	H0001010119
B	DX2 (P)	322220076790111	H0004040120
B	DX2 (Q)	322220081820111	H0002020121
B	DX2 (R)	310100037470111	H0011110122
B	DX2 (S)	310100049490011	H0001010123
B	DX2 (T)	310100052550111	H0004040124
B	DX2 (U)	310100057610111	H0005050125
B	DX2 (V)	310100064670111	H0004040126
B	DISPOSITIONS	308080041170211	H0013130127

//EOF0519

910 OCCUPATIONAL THERAPY REPEAT

SCAN910.FIL

H02830247NY00000T REPEAT

*

H0539

*

B	LITHO	739160104040011	H0024080001
B	SSN	1443601756601014	V0090090002
B	FMP	1474601756601014	V0020020003
B	OTHER REF 1	220200082660209	V0009000004
B	OTHER REF 2	2161600826602024	V0009020005
B	REF CODE PFX	2111100786802024	V0006010006
B	REF CODE (1A)	009090079660109	V0014000007
B	REF CODE (1B)	0080800786601024	V0013010008
B	REF CODE (2A)	006060079660109	V0014000009
B	REF CODE (2B)	0050500786601024	V0013010010
B	REF CODE (3A)	003030079660109	V0014000011
B	REF CODE (3B)	0020200786601024	V0013010012
B	DX PFX	2343400757302024	V0002010013
B	DX	1332901756601034	V0050050014
B	DATE (MO) 1	2424200615001014	V0012020015
B	DATE (DY) 1	1464402595001014	V0020020016
B	PROV1 PFX 1A	024240059460109	V0014000017
B	PROV1 PFX 1B	0222200584601014	V0013010018
B	PROV1 CODE 1	1201402595001014	V0040040019
B	PROV1 TIME 1A	220200048450109	V0004000020
B	PROV1 TIME 1B	2161600484501014	V0004010021
B	PROV2 PFX 1A	012120059460109	V0014000022
B	PROV2 PFX 1B	0101000584601024	V0013010023
B	PROV2 CODE 1	1080202595001024	V0040040024
B	PROV2 TIME 1A	208080048450109	V0004000025
B	PROV2 TIME 1B	2040400484501024	V0004010026
B	REASON FOR #2 1A	208080041390109	V0003000027
B	REASON FOR #2 1B	206020237370009	H0003000028
B	REASON FOR #2 1C	2080800424200024	V0001010029
B	VISIT COUNT 1A	224240038360109	V0003000030
B	VISIT COUNT 1B	222220038360109	V0003000031
B	VISIT COUNT 1C	2202000383601024	V0003010032
B	UNLISTED PROC 1	1463802453601024	V0050050033
B	PROC PROV1 1	336360061360111	H0026260034
B	PROC PROV2 1	334340061360111	H0026260035
B	DATE (MO) 2	2424200312001024	V0012020036
B	DATE (DY) 2	1464402292001024	V0020020037
B	PROV1 PFX 2A	024240029160109	V0014000038
B	PROV1 PFX 2B	0222200281601024	V0013010039
B	PROV1 CODE 2	1201402292001024	V0040040040
B	PROV1 TIME 2A	220200018150109	V0004000041
B	PROV1 TIME 2B	2161600181501024	V0004010042
B	PROV2 PFX 2A	012120029160109	V0014000043
B	PROV2 PFX 2B	0101000281601024	V0013010044
B	PROV2 CODE 2	1080202292001024	V0040040045
B	PROV2 TIME 2	208080018150109	V0004000046
B	PROV2 TIME 2A	2040400181501024	V0004010047
B	REAS FOR #2 2A	208080011090109	V0003000048
B	REAS FOR #2 2B	206020207070009	H0003000049
B	REAS FOR #2 2C	2080800121200024	V0001010050
B	VISIT COUNT 2A	224240008060109	V0003000051

B VISIT COUNT 2B 222220008060109 V0003000052
 B VISIT COUNT 2C 2202000080601024V0003010053
 B UNLISTED PROC 2 1463802150601024V0050050054
 B PROC PROV1 2 336360031060111 H0026260055
 B PROC PROV2 2 334340031060111 H0026260056
 B DATE (MO) 3 2434300092001024V0012020057
 B DATE (DY) 3 1474502112001024V0020020058
 B PROV1 PFX 3A 025250011240109 V0014000059
 B PROV1 PFX 3B 0232300122401024V0013010060
 B PROV1 CODE 3 1211502112001024V0040040061
 B PROV1 TIME 3A 221210022250109 V0004000062
 B PROV1 TIME 3B 2171700222501024V0004010063
 B PROV2 PFX 3A 013130011240109 V0014000064
 B PROV2 PFX 3B 0111100122401024V0013010065
 B PROV2 CODE 3 1090302112001024V0040040066
 B PROV2 TIME 3A 209090022250109 V0004000067
 B PROV2 TIME 3B 2050500222501024V0004010068
 B REASON FOR #2 3A 209090029310109 V0003000069
 B REASON FOR #2 3B 207030233330009 H0003000070
 B REASON FOR #2 3C 2090900282800024V0001010071
 B VISIT COUNT 3A 225250032340109 V0003000072
 B VISIT COUNT 3B 223230032340109 V0003000073
 B VISIT COUNT 3C 2212100323401024V0003010074
 B UNLISTED PROC 3 1473902253401024V0050050075
 B PROC PROV1 3 337370009340111 H0026260076
 B PROC PROV2 3 335350009340111 H0026260077
 B DATE (MO) 4 2434300395001024V0012020078
 B DATE (DY) 4 1474502415001024V0020020079
 B PROV1 PFX 4A 025250041540109 V0014000080
 B PROV1 PFX 4B 0232300425401024V0013010081
 B PROV1 CODE 4 1211502415001024V0040040082
 B PROV1 TIME 4A 221210052550109 V0004000083
 B PROV1 TIME 4B 2171700525501024V0004010084
 B PROV2 PFX 4A 013130041540109 V0014000085
 B PROV2 PFX 4B 0111100425401024V0013010086
 B PROV2 CODE 4 1090302415001024V0040040087
 B PROV2 TIME 4A 209090052550109 V0004000088
 B PROV2 TIME 4B 2050500525501024V0004010089
 B REASON FOR #2 4A 209090059610109 V0003000090
 B REASON FOR #2 4B 207030263630009 H0003000091
 B REASON FOR #2 4C 2090900585800024V0001010092
 B VISIT COUNT 4A 225250062640109 V0003000093
 B VISIT COUNT 4B 223230062640109 V0003000094
 B VISIT COUNT 4C 2212100626401024V0003010095
 B UNLISTED PROC 4 1473902556401024V0050050096
 B PROC PROV1 4 337370039640111 H0026260097
 B PROC PROV2 4 335350039640111 H0026260098
 //EOF0331

H02830247NY0000PT REPEAT

H0542

B	LITHO	739160104040011	H0024080001
B	SSN	1443601756601014	V0090090002
B	FMP	1474601756601014	V0020020003
B	OTHER REF 1	220200082660209	V0009000004
B	OTHER REF 2	2161600826602024	V0009020005
B	REF CODE PFX	2111100786802024	V0006010006
B	REF CODE (1A)	009090079660109	V0014000007
B	REF CODE (1B)	0080800786601024	V0013010008
B	REF CODE (2A)	006060079660109	V0014000009
B	REF CODE (2B)	0050500786601024	V0013010010
B	REF CODE (3A)	003030079660109	V0014000011
B	REF CODE (3B)	0020200786601024	V0013010012
B	DX PFX	2343400757302024	V0002010013
B	DX	1332901756601034	V0050050014
B	DATE (MO) 1	2424200615001014	V0012020015
B	DATE (DY) 1	1464402595001014	V0020020016
B	PROV1 PFX 1A	024240059460109	V0014000017
B	PROV1 PFX 1B	0222200584601014	V0013010018
B	PROV1 CODE 1	1201402595001014	V0040040019
B	PROV1 TIME 1A	220200048450109	V0004000020
B	PROV1 TIME 1B	2161600484501014	V0004010021
B	PROV2 PFX 1A	012120059460109	V0014000022
B	PROV2 PFX 1B	0101000584601024	V0013010023
B	PROV2 CODE 1	1080202595001024	V0040040024
B	PROV2 TIME 1A	208080048450109	V0004000025
B	PROV2 TIME 1B	2040400484501024	V0004010026
B	REASON FOR #2 1A	208080041390109	V0003000027
B	REASON FOR #2 1B	206020237370009	H0003000028
B	REASON FOR #2 1C	2080800424200024	V0001010029
B	VISIT COUNT 1A	224240038360109	V0003000030
B	VISIT COUNT 1B	222220038360109	V0003000031
B	VISIT COUNT 1C	2202000383601024	V0003010032
B	UNLISTED PROC 1	1463802453601024	V0050050033
B	PROC PROV1 1	336360061360111	H0026260034
B	PROC PROV2 1	334340061360111	H0026260035
B	DATE (MO) 2	2424200312001024	V0012020036
B	DATE (DY) 2	1464402292001024	V0020020037
B	PROV1 PFX 2A	024240029160109	V0014000038
B	PROV1 PFX 2B	0222200281601024	V0013010039
B	PROV1 CODE 2	1201402292001C	4V0040040040
B	PROV1 TIME 2A	220200018150109	V0004000041
B	PROV1 TIME 2B	2161600181501024	V0004010042
B	PROV2 PFX 2A	012120029160109	V0014000043
B	PROV2 PFX 2B	0101000281601024	V0013010044
B	PROV2 CODE 2	1080202292001024	V0040040045
B	PROV2 TIME 2	208080018150109	V0004000046
B	PROV2 TIME 2A	2040400181501024	V0004010047
B	REAS FOR #2 2A	208080011090109	V0003000048
B	REAS FOR #2 2B	206020207070009	H0003000049
B	REAS FOR #2 2C	2080800121200024	V0001010050
B	VISIT COUNT 2A	224240008060109	V0003000051

B VISIT COUNT 2B 222220008060109 V0003000052
 B VISIT COUNT 2C 2202000080601024V0003010053
 B UNLISTED PROC 2 1463802150601024V0050050054
 B PROC PROV1 2 336360031060111 H0026260055
 B PROC PROV2 2 334340031060111 H0026260056
 B DATE (MO) 3 2434300092001024V0012020057
 B DATE (DY) 3 1474502112001024V0020020058
 B PROV1 PFX 3A 025250011240109 V0014000059
 B PROV1 PFX 3B 0232300122401024V0013010060
 B PROV1 CODE 3 1211502112001024V0040040061
 B PROV1 TIME 3A 221210022250109 V0004000062
 B PROV1 TIME 3B 2171700222501024V0004010063
 B PROV2 PFX 3A 013130011240109 V0014000064
 B PROV2 PFX 3B 0111100122401024V0013010065
 B PROV2 CODE 3 1090302112001024V0040040066
 B PROV2 TIME 3A 209090022250109 V0004000067
 B PROV2 TIME 3B 2050500222501024V0004010068
 B REASON FOR #2 3A 209090029310109 V0003000069
 B REASON FOR #2 3B 207030233330009 H0003000070
 B REASON FOR #2 3C 2090900282800024V0001010071
 B VISIT COUNT 3A 225250032340109 V0003000072
 B VISIT COUNT 3B 223230032340109 V0003000073
 B VISIT COUNT 3C 2212100323401024V0003010074
 B UNLISTED PROC 3 1473902253401024V0050050075
 B PROC PROV1 3 337370009340111 H0026260076
 B PROC PROV2 3 335350009340111 H0026260077
 B DATE (MO) 4 2434300395001024V0012020078
 B DATE (DY) 4 1474502415001024V0020020079
 B PKOV1 PFX 4A 025250041540109 V0014000080
 B PROV1 PFX 4B 0232300425401024V0013010081
 B PROV1 CODE 4 1211502415001024V0040040082
 B PROV1 TIME 4A 221210052550109 V0004000083
 B PROV1 TIME 4B 2171700525501024V0004010084
 B PROV2 PFX 4A 013130041540109 V0014000085
 B PROV2 PFX 4B 0111100425401024V0013010086
 B PROV2 CODE 4 1090302415001024V0040040087
 B PROV2 TIME 4A 209090052550109 V0004000088
 B PROV2 TIME 4B 2050500525501024V0004010089
 B REASON FOR #2 4A 209090059610109 V0003000090
 B REASON FOR #2 4B 207030263630009 H0003000091
 B REASON FOR #2 4C 2090900585800024V0001010092
 B VISIT COUNT 4A 225250062640109 V0003000093
 B VISIT COUNT 4B 223230062640109 V0003000094
 B VISIT COUNT 4C 2212100626401024V0003010095
 B UNLISTED PROC 4 1473902556401024V0050050096
 B PROC PROV1 4 337370039640111 H0026260097
 B PROC PROV2 4 335350039640111 H0026260098
 //EOF0331

930 REPEAT FORM

SCAN930.FIL

H02830247NY0051REPEAT PROCEDURE

*

H0144

B	LITHO	739160104040011	H0024080001
B	CLINIC PFX	2111100766802024	V0005010002
B	CLINIC CODE1	0101000776601024	V0012010003
B	CLINIC CODE2	0090801775101024	V0054020004
B	OTHER UCA	2151500785802024	V0011020005
F	TODAY'S DATE	804	040006
F	PROV1 PFX	801	010007
F	PROV1 CODE	804	040008
B	SSN	1433501635401014	V0090090009
B	FMP	1464501635401014	V0020020010
B	VISIT COUNT1	215150046440109	V0003000011
B	VISIT COUNT2	213130046440109	V0003000012
B	VISIT COUNT3	2111100464401024	V0003010013
B	PROV TIME	2232300634301014	V0021020014
B	REF CODE PFX	2050500766602024	V0006010015
B	REF CODE 1	0040400776601024	V0012010016
B	REF CODE 2	0030201775101024	V0054020017
B	PROC CODE	1242001776801014	V0050050018
B	DX PFX	2323200636102024	V0002010019
B	DX CODE	1312701635401034	V0050050020
B	DATE (MO) A	2434300362501014	V0012020021
B	DATE (DY) A1	1454500343101014	V0004010022
B	DATE (DY) A2	1444400342501014	V0010010023
B	PROV PFX (1) A	045450020070109	V0014000024
B	PROV PFX (2) A	0444400190701014	V0013010025
B	PROV CODE A	1434001201101014	V0040040026
B	DATE (MO) B	2343400362501024	V0012020027
B	DATE (DY) B1	1363600343101024	V0004010028
B	DATE (DY) B2	1353500342501024	V0010010029
B	PROV PFX (1) B	036360020070109	V0014000030
B	PROV PFX (2) B	0353500190701024	V0013010031
B	PROV CODE B	1343101201101024	V0040040032
B	DATE (MO) C	2252500362501024	V0012020033
B	DATE (DY) C1	1272700343101024	V0004010034
B	DATE (DY) C2	1262600342501024	V0010010035
B	PROV PFX (1) C	027270020070109	V0014000036
B	PROV PFX (2) C	0262600190701024	V0013010037
B	PROV CODE C	1252201201101024	V0040040038
B	DATE (MO) D	2161600362501024	V0012020039
B	DATE (DY) D1	1181800343101024	V0004010040
B	DATE (DY) D2	1171700342501024	V0010010041
B	PROV PFX (1) D	018180020070109	V0014000042
B	PROV PFX (2) D	0171700190701024	V0013010043
B	PROV CODE D	1161301201101024	V0040040044
B	DATE (MO) E	2070700362501024	V0012020045
B	DATE (DY) E1	1090900343101024	V0004010046
B	DATE (DY) E2	1080800342501024	V0010010047
B	PROV PFX (1) E	009090020070109	V0014000048
B	PROV PFX (2) E	0080800190701024	V0013010049
B	PROV CODE E	1070401201101024	V0040040050

//EOF0091

940 SHORT FORM

SCAN940.FIL

H02830247NY0051SHORT FORM

*

H0543

*

B	LITHO	739160104040011	H00240800C1
B	CLINIC PREFIX	2323200787002014	V0005010002
B	CLINIC CODE 1	0303000796801014	V0012010003
B	CLINIC CODE 2	0282602795301014	V0054020004
B	DATE (MO)	2383800604901014	V0012020005
B	DATE (DY)1	1424200585501014	V0004010006
B	DATE (DY)2	1404000584901014	V0010010007
B	PROV PFX	0464600795301014	V0027010008
B	PROV CODE	1443802786901014	V0040040009
B	SSN A	1422602362701014	V0090090010
B	FMP A	1464402362701014	V0020020011
B	TIME 1A	212120043410109	V0003000012
B	TIME 2A	208080043410109	V0003000013
B	TIME 3A	2040400434101014	V0003010014
B	UNL PROC A	1100202362701024	V0050050015
B	PROC A	324240043280111	H0016160016
B	SP PRE CLIN A	322140126260011	V0009090017
B	SSN B	1422602160701024	V0090090018
B	FMP B	1464402160701024	V0020020019
B	TIME 1B	212120023210109	V0003000020
B	TIME 2B	208080023210109	V0003000021
B	TIME 3B	2040400232101024	V0003010022
B	UNL PROC B	1100202160701024	V0050050023
B	PROC B	324240023080111	H0016160024
B	SP PRE CLIN B	322140106060011	V0009090025
B	SSN C	1432702132201024	V0090090026
B	FMP C	1474502132201024	V0020020027
B	TIME 1C	213130006080109	V0003000028
B	TIME 2C	209090006080109	V0003000029
B	TIME 3C	2050500060801024	V0003010030
B	UNL PROC C	1110302132201024	V0050050031
B	PROC C	325250006210111	H0016160032
B	SP PRE CLIN C	323150123230011	V0009090033
B	SSN D	1432702334201024	V0090090034
B	FMP D	1474502334201024	V0020020035
B	TIME 1D	213130026280109	V0003000036
B	TIME 2D	209090026280109	V0003000037
B	TIME 3D	2050500262801024	V0003010038
B	UNL PROC D	1110302334201024	V0050050039
B	PROC D	325250026410111	H0016160040
B	SP PRE CLIN D	323150143430011	V0009090041
B	SSN E	1432702536201024	V0090090042
B	FMP E	1474502536201024	V0020020043
B	TIME 1E	213130046480109	V0003000044
B	TIME 2E	209090046480109	V0003000045
B	TIME 3E	2050500464801024	V0003010046
B	UNL PROC E	1110302536201024	V0050050047
B	PROC E	325250046610111	H0016160048
B	SP PRE CLIN E	323150163630011	V0009090049

//EOF0231

950 SOCIAL WORK SHORT FORM

SCAN950.FIL

H02830247NY0051SOCIAL WORK SHORT

*

H0545

*

B	LITHO	739160104040011	H0024080001
F	OTHER UCA	801	010002
B	DATE (MO)	2424200776601014V0012020003	
B	DATE (DY) 1	1464600757201014V0004010004	
B	DATE (DY) 2	1444400756601014V0010010005	
B	PROV1 PFX1	035350077640109	V0014000006
B	PROV1 PFX2	0343400766401014V0013010007	
B	PROV1 CODE	1322602776801014V0040040008	
B	SSN A	1422602534401014V0090090009	
B	FMP A	1464402534401014V0020020010	
B	REF CODE PFX A	2464600372902024V0005010011	
B	REF CODE A	0444002381201024V0081030012	
B	TIME 1 A	236360012090109	V0004000013
B	TIME 2 A	2303000120901014V0004010014	
B	UNL PROC A	1362802241501024V0050050015	
B	PROBLEM A	2363600413601014V0006010016	
B	PROCEDURE A	336360033290111	H0005050017
B	SP PRE CLIN A	346380106060011	V0009090018
B	SSN B	1180202534401024V0090090019	
B	FMP B	1222002534401024V0020020020	
B	REF CODE PFX B	2222200372902024V0005010021	
B	REF CODE B	0201602381201024V0081030022	
B	TIME 1 B	212120012090109	V0004000023
B	TIME 2 B	2060600120901024V0004010024	
B	UNL PROC B	1120402241501024V0050050025	
B	PROBLEM B	2121200413601024V0006010026	
B	PROCEDURE B	312120033290111	H0005050027
B	SP PRE CLIN B	322140106060011	V0009090028
B	SSN C	1432702162501024V0090090029	
B	FMP C	1474502162501024V0020020030	
B	REF CODE PFX C	2474700324002024V0005010031	
B	REF CODE C	0454102315701024V0081030032	
B	TIME 1 C	237370057600109	V0004000033
B	TIME 2 C	2313100576001024V0004010034	
B	UNL PROC C	1372902455401024V0050050035	
B	PROBLEM C	2373700283301024V0006010036	
B	PROCEDURE C	337370036400111	H0005050037
B	SP PRE CLIN C	347390163630011	V0009090038
B	SSN D	1190302162501024V0090090039	
B	FMP D	1232102162501024V0020020040	
B	REF CODE PFX D	2232300324002024V0005010041	
B	REF CODE D	0211702315701024V0081030042	
B	TIME 1 D	213130057600109	V0004000043
B	TIME 2 D	2070700576001024V0004010044	
B	UNL PROC D	1130502455401024V0050050045	
B	PROBLEM D	2131300283301024V0006010046	
B	PROCEDURE D	313130036400111	H0005050047
B	SP PRE CLIN D	323150163630011	V0009090048

//EOF0162

APPENDIX B

PC BASIC SCANNER PROGRAMS

TABLE OF CONTENTS

RACP200.LST	Patient Form
RACP300.LST	Provider Form
RACPGEN.LST	General Form (400-790)
RACP800.LST	BAS/TMC Form
RACP820.LST	EKG Form
RACP830.LST	Emergency Room Form
RACP850.LST	Group I and II Forms
RACP860.LST	Immunization Short Form
RACP900.LST	Occupational Health Form
RACP910.LST	OT Repeat Form
RACP920.LST	PT Repeat Form
RACP930.LST	Repeat Form
RACP940.LST	Short Form
RACP950.LST	Social Work Short Form

PAGE 1

06-08-87

07:22:23

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

001A 0002 REM \$LINESIZE: 132
001A 0002 REM \$PAGESIZE: 66
001A 0002 REM \$TITLE: 'RACP200 '
001A 0002 REM \$SUBTITLE: 'PATIENT DESTING/DECODE PROGRAM'
001A 0002 REM \$PAGE

```

Offset  Data  Source Line
001A 0002 REM +-----+
001A 0002 REM | NAME: RXXP200          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | Date: 28 Feb 85          PATIENT REGISTRATION PROGRAM |
001A 0002 REM | Floyd Cole |
001A 0002 REM | Upd: 21 May 85 |
001A 0002 REM | Upd: 8 Jun 87 Change names for new system DRB |
001A 0002 REM +-----+
001A 0002 REM          PATIENT OMR INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the patient OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          PATIENT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RXXP05, otherwise,
001A 0002 REM the program chains to program RXXP10 on exit.
001A 0002
001A 0002 REM $INCLUDE: 'RACDIM.MOD' REM Include the DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD          DIMENSION DEFINITIONS *
001A 0002 '* Date: 28 Feb 84          Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002
001A 0002 DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 DIM SEXCODE$(2) '(DECODE FOR SEX)
001A 0002 DIM CIVCAT$(8) '(DECODE FOR CIVILIAN CATEGORY )
001A 0002 DIM ADCCAT$(3) '(DECODE FOR ACTIVE DUTY CATEGORY)
001A 0002 DIM YN$(3) '(YES/NO ANSWERS 0=?, 1 = Y , 2=N , 3=X)

```

Offset	Data	Source Line
001A	0002	DIM CAT\$(7,13) '(PATIENT CATEGORY ARRAY)
001A	0002	DIM ED.MSG\$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A	0002	DIM NOCAT\$(46) '(Other Category Array) Added 06/01/85)
001A	0002	
001A	0002	REM \$INCLUDE: 'RACCMN.MOD' REM Include the COMMON AREA DEFINITION
001A	0002	*****
001A	0002	'* NAME: RACCMN.MOD COMMON AREA DEFINITION *
001A	0002	'* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A	0002	' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' effect the changes made here.
001A	0002	'
001A	0002	' *****START OF COMMON DEFINITIONS*****
001A	0002	
001A	0002	COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL\$ 'BASIC SCREEN COLORS
001A	0002	COMMON HEADERS\$ '21 CHARACTER SCANNER HEADER INFO
001A	0002	COMMON TEXT\$ '' AINING CHARACTERS FROM SCANNER
001A	0002	COMMON PGMID\$ 'PROGRAM OR FORM ID
001A	0002	COMMON MOLENGTH() 'DAYS IN THE MONTH
001A	0002	COMMON USER\$()
001A	0002	' *****END OF COMMON DEFINITION*****
001A	0002	
001A	0002	
001A	0002	REM \$INCLUDE: 'RACDEF.MOD' REM Include the DEFAULT DEFINITIONS
001A	0002	*****
001A	0002	'* NAME: RACP01.DEF DEFAULT DEFINITIONS *
001A	0002	'* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' Variables used in common that have a default value on start*up
001A	0002	' will be held in this file. It is an included file so it cannot
001A	0002	' be run in a stand*alone mode. In normal operation, this file
001A	0002	' should be 'included' in the main program only (RACP10.BAS).
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	' *****START OF DEFAULT DEFINITION*****
001A	0002	FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
0046	1118	BACK = 1 'Background Color = Light Blue
004C	1118	BORD = 4 'BORDER = RED
0054	111A	HIDE = 4 'ALTERNATE COLOR = RED
0058	111A	EFORE= 14 'ERROR FOREGROUND DISPLAY
0062	111A	EBACK= 0 'ERROR BACKGROUND DISPLAY
0069	111A	BELL\$ = CHR\$(7) 'Sound the bell
0075	111A	
0075	111A	MOLENGTH(1) = 31 'JAN
007C	111A	MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
0083	111A	MOLENGTH(3) = 31 'MAR

Offset Data Source Line

```

008A 111A      MOLENGTH(4) = 30      'APR
0091 111A      MOLENGTH(5) = 31      'MAY
0098 111A      MOLENGTH(6) = 30      'JUN
009F 111A      MOLENGTH(7) = 31      'JUL
00A6 111A      MOLENGTH(8) = 31      'AUG
00AD 111A      MOLENGTH(9) = 30      'SEP
00B4 111A      MOLENGTH(10) = 31     'OCT
00B8 111A      MOLENGTH(11) = 30     'NOV
00C2 111A      MOLENGTH(12) = 31     'DEC
00C9 111A
00C9 111A      DATEERR$(0) = " "
0002 111A      DATEERR$(1) = "INVALID MONTH"
0008 111A      DATEERR$(2) = "INVALID DAY "
00E4 111A      DATEERR$(3) = "DAY TOO LARGE FOR MONTH CODED"
00ED 111A
00ED 111A      MAXLENGTH = 80        'MAXIMUM LENGTH OF OUTPUT RECORD
00F4 111C      PADS = " "          'PAD CHARACTER FOR SHORT RECORDS
00FD 1120
00FD 1120      ' *****END OF DEFAULT DEFINITION*****
00FD 1120
00FD 1120
00FD 1120      KEY OFF
0103 1120
0103 1120      REM *****
0103 1120      REM THE FOLLOWING VARIABLES ARE UNIQUE TO EACH PROGRAM AND MUST
0103 1120      REM BE CHANGED.
0103 1120      REM *****
0103 1120      PGMTITL$ = "PATIENT REGISTRATION"
010C 1124
010C 1124      PGMID$ = "200"          'VALUE RECEIVED FROM THE SCANNER
0115 1124          'IN HEADER VARIABLE 'PROGRAMS'
0115 1124
0115 1124      DATFIL$ = "PATIENT.DAT" 'FILE TO BE INPUT TO FOCUS
011E 1128
011E 1128      REM LENGTH OF STRING RECEIVED FROM THE OMR....
011E 1128      HEADER = 21
0125 112A      RESPONSE= 93
012C 112C      RECORDLENGTH = HEADER + RESPONSE
0137 112E
0137 112E      REM *****
0137 112E
0137 112E      BTIMES=TIMES          'SCAN START TIME
0140 1132
0140 1132      REM CIVILIAN CATEGORY TABLE
0140 1132      CIVCAT$(0)="??"
0149 1132      CIVCAT$(1)="GS"
0152 1132      CIVCAT$(2)="GM"
0158 1132      CIVCAT$(3)="WG"
0164 1132      CIVCAT$(4)="WL"
0160 1132      CIVCAT$(5)="WS"
0176 1132      CIVCAT$(6)="NF"
017F 1132      CIVCAT$(7)="CT"
0188 1132      CIVCAT$(8)="OT"
0191 1132

```

Offset Data Source Line

```

0191 1132 REM ARMY ACTIVE DUTY TABLE
0191 1132     ADCATS(0)="?? "
019A 1132     ADCATS(1)="O "
01A3 1132     ADCATS(2)="W "
01AC 1132     ADCATS(3)="E "
01B5 1132
01B5 1132
01B5 1132 REM *** PATIENT CATEGORY ***
01B5 1132 '   CIVILIAN/OTHER      ARMY      AIR FORCE
01B5 1132     CATS(0,0)="ERR" : CATS(1,0)="ERR" : CATS(2,0)="ERR"
01D0 1132     CATS(0,1)="ERR" : CATS(1,1)="A10" : CATS(2,1)="F10" 'AD
01EB 1132     CATS(0,2)="ERR" : CATS(1,2)="A21" : CATS(2,2)="F20" 'RES IET
0206 1132     CATS(0,3)="ERR" : CATS(1,3)="A23" : CATS(2,3)="F20" 'RES IDT/AT
0221 1132     CATS(0,4)="ERR" : CATS(1,4)="A24" : CATS(2,4)="F20" 'NG IET
023C 1132     CATS(0,5)="ERR" : CATS(1,5)="A26" : CATS(2,5)="F20" 'NG IDT/AT
0257 1132     CATS(0,6)="ERR" : CATS(1,6)="A70" : CATS(2,6)="F70" 'ACADEMY CDT
0272 1132     CATS(0,7)="ERR" : CATS(1,7)="A80" : CATS(2,7)="F80" 'ROTC CDT
0280 1132     CATS(0,8)="ERR" : CATS(1,8)="A50" : CATS(2,8)="F50" 'DEPN AD
02A8 1132     CATS(0,9)="ERR" : CATS(1,9)="A30" : CATS(2,9)="F30" 'RETIRED
02C3 1132     CATS(0,10)="ERR" : CATS(1,10)="A60" : CATS(2,10)="F60" 'DEP RET/DEC
02DE 1132     CATS(0,11)="H26" : CATS(1,11)="ERR" : CATS(2,11)="ERR" 'CIVILIAN
02F9 1132     CATS(0,12)="H26" : CATS(1,12)="ERR" : CATS(2,12)="ERR" 'CIV. PRE EMP
0314 1132     CATS(0,13)="OTH" : CATS(1,13)="OTH" : CATS(2,13)="OTH"
032F 1132
032F 1132 '   NAVY      MARINE      COAST GUARD
032F 1132     CATS(3,0)="ERR" : CATS(4,0)="ERR" : CATS(5,0)="ERR"
034A 1132     CATS(3,1)="N10" : CATS(4,1)="M10" : CATS(5,1)="C10"
0365 1132     CATS(3,2)="N20" : CATS(4,2)="M20" : CATS(5,2)="C20"
0380 1132     CATS(3,3)="N20" : CATS(4,3)="M20" : CATS(5,3)="C20"
0398 1132     CATS(3,4)="N20" : CATS(4,4)="M20" : CATS(5,4)="C20"
03B6 1132     CATS(3,5)="N20" : CATS(4,5)="M20" : CATS(5,5)="C20"
03D1 1132     CATS(3,6)="N70" : CATS(4,6)="ERR" : CATS(5,6)="C70" 'ACADEMY CDT
03EC 1132     CATS(3,7)="N80" : CATS(4,7)="ERR" : CATS(5,7)="ERR" 'ROTC CDT
0407 1132     CATS(3,8)="N50" : CATS(4,8)="M50" : CATS(5,8)="C50"
0422 1132     CATS(3,9)="N30" : CATS(4,9)="M30" : CATS(5,9)="C30"
0430 1132     CATS(3,10)="N60" : CATS(4,10)="M60" : CATS(5,10)="C60"
0458 1132     CATS(3,11)="ERR" : CATS(4,11)="ERR" : CATS(5,11)="ERR"
0473 1132     CATS(3,12)="ERR" : CATS(4,12)="ERR" : CATS(5,12)="ERR"
048E 1132     CATS(3,13)="OTH" : CATS(4,13)="OTH" : CATS(5,13)="OTH"
04A9 1132
04A9 1132 '   USPHS      OTHER
04A9 1132     CATS(6,0)="ERR" : CATS(7,0)="OTH"
04BB 1132     CATS(6,1)="P10" : CATS(7,1)="ERR"
04CD 1132     CATS(6,2)="P20" : CATS(7,2)="ERR"
04DF 1132     CATS(6,3)="P20" : CATS(7,3)="ERR"
04F1 1132     CATS(6,4)="P20" : CATS(7,4)="ERR"
0503 1132     CATS(6,5)="P20" : CATS(7,5)="ERR"
0515 1132     CATS(6,6)="ERR" : CATS(7,6)="ERR"
0527 1132     CATS(6,7)="ERR" : CATS(7,7)="ERR"
0539 1132     CATS(6,8)="P50" : CATS(7,8)="ERR"
054B 1132     CATS(6,9)="P30" : CATS(7,9)="ERR"
055D 1132     CATS(6,10)="P60" : CATS(7,10)="ERR"
056F 1132     CATS(6,11)="ERR" : CATS(7,11)="ERR"
0581 1132     CATS(6,12)="ERR" : CATS(7,12)="ERR"

```

Offset Data Source Line

```
0593 1132 CAT$(6,13)="OTH" : CAT$(7,13)="OTH"
05A5 1132
05A5 1132 REM *** Other Patient Category Table ***
05A5 1132 NOCAT$(00) = "00ERR"
05AE 1132 NOCAT$(01) = "13A10"
05B7 1132 NOCAT$(02) = "44010"
05C0 1132 NOCAT$(03) = "45020"
05C9 1132 NOCAT$(04) = "46030"
05D2 1132 NOCAT$(05) = "47050"
05DB 1132 NOCAT$(06) = "48060"
05E4 1132 NOCAT$(07) = "49H10"
05ED 1132 NOCAT$(08) = "50H20"
05F6 1132 NOCAT$(09) = "52H30"
05FF 1132 NOCAT$(10) = "53H38"
0608 1132 NOCAT$(11) = "54H40"
0611 1132 NOCAT$(12) = "55H50"
061A 1132 NOCAT$(13) = "56J10"
0623 1132 NOCAT$(14) = "57J20"
062C 1132 NOCAT$(15) = "58J30"
0635 1132 NOCAT$(16) = "59K10"
063E 1132 NOCAT$(17) = "60K20"
0647 1132 NOCAT$(18) = "61K30"
0650 1132 NOCAT$(19) = "62K40"
0659 1132 NOCAT$(20) = "63K42"
0662 1132 NOCAT$(21) = "64K43"
066B 1132 NOCAT$(22) = "65K50"
0674 1132 NOCAT$(23) = "67K50"
067D 1132 NOCAT$(24) = "68K51"
0686 1132 NOCAT$(25) = "69K60"
068F 1132 NOCAT$(26) = "70K70"
0698 1132 NOCAT$(27) = "71Q10"
06A1 1132 NOCAT$(28) = "72R10"
06AA 1132 NOCAT$(29) = "73S10"
06B3 1132 NOCAT$(30) = "74S20"
06BC 1132 NOCAT$(31) = "75S23"
06C5 1132 NOCAT$(32) = "76S24"
06CE 1132 NOCAT$(33) = "77S30"
06D7 1132 NOCAT$(34) = "78S32"
06E0 1132 NOCAT$(35) = "79S40"
06E9 1132 NOCAT$(36) = "80S50"
06F2 1132 NOCAT$(37) = "81S60"
06FB 1132 NOCAT$(38) = "82X10"
0704 1132 NOCAT$(39) = "83X20"
070D 1132 NOCAT$(40) = "84X30"
0716 1132 NOCAT$(41) = "85X40"
071F 1132 NOCAT$(42) = "86X45"
0728 1132 NOCAT$(43) = "87X52"
0731 1132 NOCAT$(44) = "88X60"
073A 1132 NOCAT$(45) = "89X70"
0743 1132
0743 1132 REM SEX TABLE
0773 1132 SEXCODE$(0)="7" : SEXCODE$(1)="M" : SEXCODE$(2)="F"
075E 1132
075E 1132 REM YES/NO TABLE
```

RACP200

PATIENT DESTING/DECODE PROGRAM

PAGE 7

06-C8-87

07:22:23

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

075E 1132 YNS(0)="?" : YNS(1)="Y" : YNS(2)="N" : YNS(3)="X"

0782 1132

0782 1132 REM \$PAGE

Offset	Data	Source Line
0782	1132	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
0787	1132	CLS
0788	1132	GOSUB 7000 'PRINT SCREEN HEADING
0790	1132	
0790	1132	REM *****
0790	1132	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
0790	1132	REM *****
0790	1132	REM
0790	1132	OPEN DATFIL\$ FOR APPEND AS #1
07A2	1132	
07A2	1132	REM *****
07A2	1132	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
07A2	1132	REM *****
07A2	1132	REM LPRINT CHR\$(15); 'INCLUDE FOR NARROW PART PAPER
07A2	1132	WIDTH "LPT1:",160
07AC	1132	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
07B8	1134	COLOR 14
07BF	1134	LOCATE 11,26 : PRINT "PATIENT REGISTRATION FORM"
07D4	1134	COLOR FORE,BACK,BORD
07EA	1134	
07EA	1134	REM *****
07EA	1134	REM **** COMMUNICATIONS SETUP ****
07EA	1134	REM *****
07EA	1134	REM PROTOCOL
07EA	1134	GOSUB 9001
07EF	1134	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0809	1138	
0809	1138	REM START SCANNER (S1)
0809	1138	CNTRLOPT =1 :GOSUB 9010
0815	113A	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
082F	113A	
082F	113A	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
0844	113A	READYTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
0848	113C	
0848	113C	REM *****
0848	113C	REM **** SET SCAN SHEET CALL ****
0848	113C	REM *****
0848	113C	REM
0848	113C	
0848	113C	10 REM - RETURN POINT TO READ NEXT SHEET
084C	113C	
084C	113C	AS=INKEY\$
0855	1140	IF AS=CHR\$(27) THEN GOTO 25000
0868	1140	
0868	1140	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
0870	1140	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
088C	1144	
088C	1144	TEXT\$="" 'CLEAR THE INPUT AREA
0893	1144	GOSUB 8000 'DECODE HEADER
089A	1144	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
089F	1144	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
09A4	1144	LITHO\$ = MID\$(TEXT\$,39,8)
09B6	1148	GOSUB 8070 'CHECK FOR SCANNER ERRORS
09B8	1148	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

Offset	Data	Source Line
08C0	1148	
08C0	1148	REM \$INCLUDE: 'RXXM201.MOD' INCLUDE THE PATIENT REFORMAT/EDIT
08C0	1148	*****
08C0	1148	REM **** AMBULATORY CARE INFORMATION SYSTEM 13 APR 85 ****
08C0	1148	REM **** SKIP COLE ****
08C0	1148	REM **** MODULE NAME : RXXM201.MOD ****
08C0	1148	REM **** SCANNER PROGRAM # : 200-PATIENT REGISTRATION ****
08C0	1148	REM ****
08C0	1148	REM **** PURPOSE : REFORMAT/EDIT THE PATIENT ****
08C0	1148	REM **** OMR RECORD. ****
08C0	1148	REM ****
08C0	1148	*****
08C0	1148	REM **** RESERVED LINE NUMBERS 100-199 ****
08C0	1148	*****
08C0	1148	
08C0	1148	N.ERR =0 'COUNTS THE NUMBER OF ERRORS
08C7	114A	
08C7	114A	REM PATIENT ID (SSN+FMP+DOB)
08C7	114A	100 FMP\$=MID\$(TEXT\$,31,2)
08D9	114E	DOB\$=MID\$(TEXT\$,33,6)
08E8	1152	
08E8	1152	'EDIT FMP
08E8	1152	110 CK.5010\$=FMP\$
08F4	1156	GOSUB 5010 'NUMERIC STRING CHECK
08F9	1156	IF RT.5010 = 0 THEN GOTO 115
0908	1158	N.ERR=N.ERR+1
0910	1158	ED.MSG\$(N.ERR)="02 PATIENT FAMILY MEMBER PREFIX"
0924	1158	
0924	1158	'EDIT DOB
0924	1158	115 CK.5000\$=DOB\$
092D	115C	GOSUB 5000 'DATE CHECK
0932	115C	
0932	115C	CK.5010\$=DOB\$
0938	115C	GOSUB 5010 'NUMERIC STRING CHECK
0940	115C	IF RT.5000 = 0 AND RT.5010 = 0 THEN GOTO 120
0964	115E	N.ERR=N.ERR+1
096C	115E	ED.MSG\$(N.ERR)="03-05 DATE OF BIRTH " + DATEERR\$(RT.5000)
0992	115E	
0992	115E	REM *** LITHO CODE ***
0992	115E	120 LITHO\$ = MID\$(TEXT\$,39,8)
09A4	115E	'EDIT LITHO
09A4	115E	CK.5010\$=LITHO\$
09AD	115E	GOSUB 5010 'NUMERIC STRING CHECK
09B2	115E	IF RT.5010 = 0 THEN GOTO 125
09C1	115E	N.ERR=N.ERR+1
09C9	115E	ED.MSG\$(N.ERR)="LITHO CODE "
09D0	115E	
09D0	115E	REM *** REGISTRATION DATE ***
09D0	115E	125 RDATE\$ = MID\$(TEXT\$,47,6)
09EF	1162	
09EF	1162	'EDIT REGISTRATION DATE
09EF	1162	CK.5000\$=RDATE\$
09F8	1162	GOSUB 5000 'DATE CHECK
09FD	1162	

RACP200
PATIENT DESTING/DECODE PROGRAM

PAGE 10
06-08-87
07:22:23

IBM Personal Computer BASII Compiler V1.00

Offset	Date	Source Line
09FD	1162	CK.5010\$=RDATE\$
0A06	1162	GOSUB 5010 'NUMERIC STRING CHECK
0A08	1162	IF RT.5000 = 0 AND RT.5010 = 0 THEN GOTO 130
0A2F	1162	N.ERR=N.ERR+1
0A37	1162	ED.MSG\$(N.ERR)="06-08 REGISTRATION DATE " + DATEERR\$(RT.5000)
0A50	1162	
0A50	1162	REM *** GENDER ***
0A50	1162	130 X=VAL(MID\$(TEXT\$,53,1))
0A73	1164	GENDER\$=SEXCODE\$(X)
0A85	1168	
0A85	1168	REM *** RACE ***
0A85	1168	135 RACE\$=MID\$(TEXT\$,54,1)
0A97	116C	
0A97	116C	REM *** VA ***
0A97	116C	140 X=VAL(MID\$(TEXT\$,55,1))
0AAD	116C	VAS=YNS(X)
0ABF	1170	
0ABF	1170	REM *** HCI ***
0ABF	1170	145 X=VAL(MID\$(TEXT\$,56,1))
0AD5	1170	HCI\$=YNS(X)
0AE7	1174	
0AE7	1174	REM *** TDY ***
0AE7	1174	150 X=VAL(MID\$(TEXT\$,57,1))
0AFD	1174	TDY\$=YNS(X)
0B0F	1178	
0B0F	1178	REM *** ZIP ***
0B0F	1178	155 ZIP\$=MID\$(TEXT\$,58,5)
0B21	117C	
0B21	117C	REM *** PATIENT CATEGORY ***
0B21	117C	160 STATUS\$=MID\$(TEXT\$,63,13) 'GET STATUS VALUE
0B33	1180	STATUS=INSTR(STATUS\$,"1")
0B41	1182	BRANCH=VAL(MID\$(TEXT\$,76,1)) 'GET BRANCH VALUE
0B57	1184	OTHER\$=MID\$(TEXT\$,110,2)
0B69	1188	
0B69	1188	STAT\$=CAT\$(BRANCH,STATUS)
0B85	118C	
0B85	118C	IF STAT\$ = "ERR" THEN GOTO 170 'INVALID COMBINATION
0B97	118C	IF STAT\$ = "OTH" THEN GOTO 165
0BA9	118C	GOTO 175
0BAD	118C	
0BA0	118C	REM CHECK FOR OTHER CODING
0CAD	118C	165 STAT\$ = "ERR"
04B6	118C	LET X=0
0BB0	118C	167 LET X=X+1
0BC5	118C	IF X>45 THEN GOTO 170
0BD4	118C	TEMPCAT\$=LEFT\$(NOCAT\$(X),2)
0BEC	1190	IF TEMPCAT\$ > OTHER\$ THEN GOTO 170
0BFE	1190	IF TEMPCAT\$ < OTHER\$ THEN GOTO 167
0C10	1190	STAT\$=RIGHT\$(NOCAT\$(X),3)
0C28	1190	GOTO 175
0C2C	1190	
0C2C	1190	170 REM INVALID COMBINATION CODED
0C2D	1190	N.ERR=N.ERR+1
0C35	1190	ED.MSG\$(N.ERR)="15-16 INVALID BRANCH/STATUS/OTHER CATEGORY "

RACP200

PATIENT DESTING/DECODE PROGRAM

PAGE 11

06-08-87

07:22:23

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```

0C49 1190
0C49 1190 'EDIT SSN
0C49 1190 175 IF STATUS < 11 THEN SSNS=MID$(TEXT$,22,9) : GOTO 177
0C6A 1194 IF STATUS = 13 THEN SSNS=MID$(TEXT$,22,9) : GOTO 177
0C88 1194 SSNS=MID$(TEXT$,90,9)
0C90 1194
0C90 1194 177 PTIDS = SSNS+FMP$+DOB$
0CB1 1198
0CB1 1198 CK.5010$=SSNS
0CBA 1198 GOSUB 5010 'NUMERIC STRING CHECK
0CBF 1198 IF RT.5010 = 0 THEN GOTO 178
0CCE 1198 N.ERR=N.ERR+1
0CD6 1198 ED.MSG$(N.ERR)="01 22 PATIENT SOCIAL SECURITY NUMBER"
0CEA 1198
0CEA 1198
0CEA 1198 REM *** FOREIGN STUDENT CODE ***
0CEA 1198 178 FOREIGN$= MID$(TEXT$,112,3)
0CFC 119C IF FOREIGN$ = " " GOTO 190
0DOE 119C IF FOREIGN$ = "000" GOTO 180
0D20 119C IF FOREIGN$ > "227" GOTO 179
0D32 119C IF OTHER$ > "68" AND OTHER$ < "78" GOTO 180 'GOOD PAIRING
0D59 119C
0D59 119C 179 N.ERR=N.ERR+1
0D61 119C ED.MSG$(N.ERR)= "28 FOREIGN NATIONAL CODED .. NOT ALLOWED WITH THIS CATEGORY"
0D75 119C
0D75 119C 180 REM --END OF FOREIGN NATIONAL CHECK
0D76 119C
0D76 119C REM *** AD ARMY PREFIX ***
0D76 119C X = VAL(MID$(TEXT$,77,1))
0D8C 119C MILPFX$= ADCAT$(X)
0D9E 11A0
0D9E 11A0 REM *** AD ARMY PAY GRADE ***
0D9E 11A0 MILGRADES= MID$(TEXT$,78,2)
0D80 11A4
0D80 11A4 REM *** AD ARMY MOS ***
0D80 11A4 MILMOSS= MID$(TEXT$,86,3)
0DC2 11A8
0DC2 11A8 REM *** MILITARY SSI ***
0DC2 11A8 MILSSI$= MID$(TEXT$,89,1)
0DD4 11AC
0DD4 11AC REM *** MILITARY UNIT ***
0DD4 11AC MILUNITS= MID$(TEXT$,80,6)
0DE6 11B0
0DE6 11B0 REM *** FEDERAL PREFIX ***
0DE6 11B0 X = VAL(MID$(TEXT$,99,1))
0DFC 11B0 FEDPFX$= CIVCAT$(X)
0E0E 11B4
0E0E 11B4 REM *** FEDERAL PAY GRADE ***
0E0E 11B4 FEDGRADES= MID$(TEXT$,100,2)
0E20 11B8
0E20 11B8 REM *** FEDERAL JOB SERIES ***
0E20 11B8 FEDJOBDS= MID$(TEXT$,102,4)
0E32 11B8
0E32 11B8 REM *** FEDERAL BUILDING NO. ***

```

Offset Date Source Line

```

0E32 11BC          FEDBLDGS= MID$(TEXTS,106,4)
0E44 11C0
0E44 11C0  REM   *** DUAL SSN NUMBER   ***
0E44 11C0          DUALSSNS= MID$(TEXTS,90,9)
0E56 11C4
0E56 11C4  REM   *** ACTIVE DUTY ARMY EDITS ***
0E56 11C4  IF STATS <> "A10" GOTO 185
0E68 11C4      IF MILPFX$ = " " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="17 MILITARY PREFIX"
0E92 11C4      IF MILGRADES=" " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="18 MILITARY GRADE "
0EBC 11C4      IF MILMOSS = " " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="20 MILITARY MOS  "
0EE6 11C4      IF MILSSIS = " " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="21 MILITARY SSI  "
0F10 11C4      IF MILUNITS = " " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="19 MILITARY UIC "
0F3A 11C4      IF MILPFX$ = "W " THEN GOTO 199
0F4C 11C4      IF VAL(MILMOSS)>100 THEN N.ERR=N.ERR+1 : ED.MSG$(N.ERR)="20 INVALID MOS FOR OFFICER/ENLISTED"
0F79 11C4      GOTO 199
0F7D 11C4
0F7D 11C4  185 REM   *** CIVILIAN EDITS ***
0F7E 11C4  IF STATS <> "H26" GOTO 199
0F90 11C4      IF FEDPFX$ = " " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="15 CIVILIAN CATEGORY"
0FBA 11C4      IF FEDGRADES=" " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="24 CIVILIAN GRADE "
0FE4 11C4      IF FEDJOBDS=" " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="25 CIVILIAN SERIES"
100E 11C4      IF FEDBLDGS = " " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="26 CIVILIAN BLDG"
1038 11C4      IF DUALSSNS = " " THEN N.ERR=N.ERR+1 :ED.MSG$(N.ERR)="27 CIVILIAN SSN "
1062 11C4
1062 11C4  199 REM
1063 11C4
1063 11C4  REM -----END OF MODULE  RXXM201.MOD-----
1063 11C4
1063 11C4      IF N.ERR = 0 THEN GOTO 997
1072 11C4      LPRINT "LITHO # ";LITHOS;" ... ERRORS"
1084 11C4      FOR I200 = 1 TO N.ERR
1091 11C6          LPRINT USING "### ";I200;
1090 11C8          LPRINT "=> ";ED.MSG$(I200)
1083 11C8      NEXT I200
10C4 11C8      LN.COUNT = LN.COUNT + N.ERR + 1
10D0 11CA      CNTRLPT = 6
10D7 11CA      GOSUB 9010          'REJECT THE FORM
10DC 11CA      GOTO 998          'BYPASS THE DISK WRITER....
10E0 11CA
10E0 11CA  997  REM $INCLUDE: 'RXXM202.MOD'      REM INCLUDE THE PATIENT DISK WRITER
10E1 11CA  REM *****
10E1 11CA  REM *****  AMBULATORY CARE INFORMATION SYSTEM      13 APR 85      *****
10E1 11CA  REM *****                                SKIP COLE      *****
10E1 11CA  REM *****  MODULE NAME      :  RXXM202.MOD      *****
10E1 11CA  REM *****  SCANNER PROGRAM #  :  200-PATIENT REGISTRATION      *****
10E1 11CA  REM *****
10E1 11CA  REM *****  PURPOSE      :  CREATE AND WRITE THE DISK      *****
10E1 11CA  REM *****                                RECORD FOR INPUT TO FOCUS      *****
10E1 11CA  REM *****
10E1 11CA  REM *****
10E1 11CA  REM *****  RESERVED LINE NUMBERS 200-299      *****
10E1 11CA  REM *****
10E1 11CA  REM *****
10E1 11CA  REM *****

```

Offset	Data	Source Line
10E1	11CA	REM BUILD THE OUTPUT RECORD
10E1	11CA	
10E1	11CA	RECOUTS="200" 'TRANSACTION IDENTIFIER
10EA	11CE	
10EA	11CE	RECOUTS= RECOUTS + PTIDS + STATS +LITHOS + RDATES + GENDERS + RACES
1115	11CE	
1115	11CE	
1115	11CE	REM *** FIND OUT IF WE HAVE A CIVILIAN, OR AD PATIENT ***
1115	11CE	
1115	11CE	IF LEFT\$(STATS,1)="H" THEN GOTO 200
112E	11CE	IF LEFT\$(STATS,1)="J" THEN GOTO 200
1147	11CE	IF LEFT\$(STATS,1)="P" THEN GOTO 200
1160	11CE	IF LEFT\$(STATS,1)="K" THEN GOTO 200
1179	11CE	IF LEFT\$(STATS,1)="X" THEN GOTO 200
1192	11CE	IF LEFT\$(STATS,1)="P" THEN GOTO 200
11AB	11CE	IF RIGHT\$(STATS,2)="50" THEN GOTO 200
11C4	11CE	IF RIGHT\$(STATS,2)="60" THEN GOTO 200 'CIVILIAN/DEPENDENT
11DD	11CE	
11DD	11CE	RECOUTS = RECOUTS + MILPFX\$ + MILGRADES + MILMOSS + MILSSIS + MILUNITS
1202	11CE	
1202	11CE	GOTO 210
1206	11CE	
1206	11CE	200 RECOUTS = RECOUTS + FEDPFX\$ + FEDGRADES + FEDJOBDS + FEDBLDG\$+" "
122B	11CE	
122B	11CE	210 RECOUTS = RECOUTS + TDYS + VAS + NCIS + ZIP\$
124A	11CE	RECOUTS = RECOUTS + DUALSSNS + FOREIGN\$
1250	11CE	
1250	11CE	PAD=MAXLENGTH - LEN(RECOUTS) 'FIND OUT HOW SHORT THE RECORD IS
126B	11DD	RECOUTS = RECOUTS + STRING\$(PAD,PAD\$) 'PAD THE RECORD WITH FILL CHAR
127E	11DD	
127E	11DD	PRINT #1,RECOUTS
1289	11DD	
1289	11DD	REM -----END OF MODULE RXXM202.MOD-----
1289	11DD	
1289	11DD	
1289	11DD	998 REM CONTINUE
128A	11DD	
128A	11DD	999 READTYPE = 2
1291	11DD	IF LN.COUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
12A1	11DD	GOTO 10
12A5	11DD	
12A5	11DD	REM END OF SCAN/DECODE/WRITE LOOP =====
12A5	11DD	
12A5	11DD	1000 REM \$INCLUDE: 'RACS1000.SUB' Include the VERIFY LOGON SUB
12A6	11DD	REM *****
12A6	11DD	REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
12A6	11DD	REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
12A6	11DD	REM *****
12A6	11DD	REM PATIENT OMR INPUT PROGRAM *
12A6	11DD	REM *
12A6	11DD	REM This program verifies user is logged on properly. If there is no *
12A6	11DD	REM valid user logged on at the time of execution, this subroutine will*
12A6	11DD	REM chain to the logon program RACP05, otherwise a return is issued. *
12A6	11DD	REM *****

Offset	Data	Source Line
12A6	11D0	REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
12A6	11D0	REM *****
12A6	11D0	1001 OPEN "I",1,"RACLOG.DAT"
12B8	11D0	IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
12C6	11D0	INPUT #1,USERS(1),DT\$,TMS,PIDS
12E7	11DC	IF USERS(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
12F5	11DC	IF USERS(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
1303	11DC	CLOSE 1
130A	11DC	SCREEN 0,1,0,0
1320	11DC	COLOR FORE,BACK,BORD
1336	11DC	CLS
133A	11DC	RETURN
133D	11DC	
133D	11DC	1002 CLOSE
1341	11DC	CHAIN "RACPO5"
1348	11DC	'=====END OF LOGON VERIFY SUBROUTINE 1000=====
1348	11DC	
1348	11DC	2000 REM \$INCLUDE: 'RACS2000.SUB' Include the REPLY/DELAY SUB
1349	11DC	REM *****
1349	11DC	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1349	11DC	REM **** SKIP COLE ****
1349	11DC	REM **** SUBROUTINE NAME : RACS2000.SUB ****
1349	11DC	REM **** SCANNER PROGRAM # : ALL ****
1349	11DC	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1349	11DC	REM **** SERVERS AS A WAIT AND REPLY ****
1349	11DC	REM **** ENTRY MODULE ****
1349	11DC	REM **** INPUT : SINGLE KEYBOARD ENTRY ****
1349	11DC	REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
1349	11DC	REM **** RESERVED LINE ****
1349	11DC	REM **** NUMBERS : 2001-2010 ****
1349	11DC	REM *****
1349	11DC	2001 REM REPLY FUNCTION
134A	11DC	2002 REPLY\$=INKEY\$: IF REPLY\$="" THEN 2002
135E	11E0	REPLY=ASC(REPLY\$)
1368	11E2	IF REPLY > 90 THEN REPLY\$=CHR\$(REPLY XOR 32) 'CONVERT TO CAPS
1383	11E2	IF REPLY\$ < "A" OR REPLY\$ > "Z" THEN REPLY\$="?"
13AF	11E2	RETURN
13B2	11E2	
13B2	11E2	5000 REM \$INCLUDE: 'RACS5000.SUB' Include the DATE EDITOR SUB
13B3	11E2	REM *****
13B3	11E2	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
13B3	11E2	REM **** SKIP COLE ****
13B3	11E2	REM **** SUBROUTINE NAME : RXXS5000.SUB ****
13B3	11E2	REM **** SCANNER PROGRAM # : ALL ****
13B3	11E2	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
13B3	11E2	REM **** PERFORMS A DATE EDIT ****
13B3	11E2	REM **** INPUT : DATE TO BE CHECKED MUST BE ****
13B3	11E2	REM **** IN THE VARIABLE NAMED ****
13B3	11E2	REM **** 'CK.5000\$' ****
13B3	11E2	REM **** IN THE FORMAT "YYMMDD" ****
13B3	11E2	REM ****

Offset	Data	Source Line
1383	11E2	REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
1383	11E2	REM **** VARIABLE. IF THIS VARIABLE ****
1383	11E2	REM **** CONTAINS ANY NUMBER OTHER ****
1383	11E2	REM **** THAN 0, AN ERROR WAS FOUND ****
1383	11E2	REM **** IN THE DATE. ****
1383	11E2	REM **** ****
1383	11E2	REM **** RESERVED LINE ****
1383	11E2	REM **** NUMBERS : 5001-5009 ****
1383	11E2	REM ****
1383	11E2	RT.5000 = 0
138A	11E2	CKYEAR = VAL(LEFT\$(CK.5000\$,2)) 'YEAR NUMERIC VALUE
13CD	11E4	CKMONTH = VAL(MID\$(CK.5000\$,3,2)) 'MONTH NUMERIC VALUE
13E3	11E6	CKDAY = VAL(RIGHT\$(CK.5000\$,2)) 'DAY NUMERIC VALUE
13F6	11E8	
13F6	11E8	IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
140C	11E8	IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
1422	11E8	IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
1438	11E8	IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
144E	11E8	
144E	11E8	REM LEAP YEAR CHECK
144E	11E8	MOLENGTH(2) = 28
1455	11E8	IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY
1464	11E8	IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR
1479	11E8	MOLENGTH(2) = 29
1480	11E8	
1480	11E8	5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
149F	11E8	
149F	11E8	5009 RETURN
14A2	11E8	
14A2	11E8	REM -----END OF SUBROUTINE 5000 -----
14A2	11E8	
14A2	11E8	5010 REM \$INCLUDE: 'RACS5010.SUB' Include the NUMERIC STRING EDITOR
14A3	11E8	REM ****
14A3	11E8	REM **** AMBULATORY CARE DATA BASE 1 MAY 85 ****
14A3	11E8	REM **** SKIP COLE ****
14A3	11E8	REM **** SUBROUTINE NAME : RXXS5010.SUB ****
14A3	11E8	REM **** SCANNER PROGRAM # : ALL ****
14A3	11E8	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
14A3	11E8	REM **** PERFORMS A NUMERIC STRING ****
14A3	11E8	REM **** EDIT. ****
14A3	11E8	REM ****
14A3	11E8	REM **** INPUT : STRING TO BE EDITED IS IN ****
14A3	11E8	REM **** THE VARIABLE NAMED ****
14A3	11E8	REM **** 'CK.5010\$' ****
14A3	11E8	REM ****
14A3	11E8	REM **** OUTPUT : 'RT.5010' IS THE RETURN CODE ****
14A3	11E8	REM **** VARIABLE. IF THIS VARIABLE ****
14A3	11E8	REM **** CONTAINS ANY NUMBER OTHER ****
14A3	11E8	REM **** THAN 0, AN ERROR WAS FOUND ****
14A3	11E8	REM **** IN THE STRING. ****
14A3	11E8	REM ****
14A3	11E8	REM **** RESERVED LINE ****
14A3	11E8	REM **** NUMBERS : 5011-5019 ****

```

Offset  Data  Source Line
14A3  11E8  REM *****
14A3  11E8          RT.5010 = 0
14AA  11E8
14AA  11E8          FOR I5010 = 1 TO LEN(CK.5010$)
14BA  11EA          J5010= ASC(MID$(CK.5010$,I5010,1))
14CE  11EE          IF J5010 < 48 OR J5010 > 57 THEN RT.5010 = RT.5010 + 1
14F6  11EE          NEXT I5010
1507  11EE
1507  11EE          RETURN
150A  11EE  REM ----- END OF SUBROUTINE 5010 -----
150A  11EE
150A  11EE  7000  REM $INCLUDE: 'RACS7000.SUB' Include the SCREEN HEADER SUB
150B  11EE  REM *****
150B  11EE  REM *****  AMBULATORY CARE DATA BASE          13 APR 85      ****
150B  11EE  REM *****                                SKIP COLE          ****
150B  11EE  REM *****  SUBROUTINE NAME      :  RACS7000.SUB          ****
150B  11EE  REM *****  SCANNER PROGRAM #    :  ALL                  ****
150B  11EE  REM *****  FUNCTION              :  THIS SUBROUTINE MODULE ****
150B  11EE  REM *****                                PRINTS THE STANDARD SCREEN ****
150B  11EE  REM *****                                HEADING.          ****
150B  11EE  REM *****  INPUT                  :  COMMON VARIABLE USERS(2) ****
150B  11EE  REM *****                                SYSTEM DATE          ****
150B  11EE  REM *****                                ****
150B  11EE  REM *****  OUTPUT                  :  SCREEN HEADING      ****
150B  11EE  REM *****                                ****
150B  11EE  REM *****  RESERVED LINE          ****
150B  11EE  REM *****  NUMBERS              :  7001-7010          ****
150B  11EE  REM *****
150B  11EE
150B  11EE  7001  LOCATE 1,1
1515  11EE          PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
151D  11EE          LOCATE 1,65
152A  11EE          PRINT DATES;
1532  11EE          LOCATE 2,1
153F  11EE          PRINT "USER : ";USERS(1)
154C  11EE          RETURN
154F  11EE  7100  REM $INCLUDE: 'RACS7100.SUB' Include the PRINTER HEADER SUB
1550  11EE  REM *****
1550  11EE  REM *****  AMBULATORY CARE DATA BASE          13 APR 85      ****
1550  11EE  REM *****                                SKIP COLE          ****
1550  11EE  REM *****  SUBROUTINE NAME      :  RXXS7100.SUB          ****
1550  11EE  REM *****  SCANNER PROGRAM #    :  ALL                  ****
1550  11EE  REM *****  FUNCTION              :  THIS SUBROUTINE MODULE ****
1550  11EE  REM *****                                PRINTS THE STANDARD HEADING ****
1550  11EE  REM *****                                ON THE PRINTER.      ****
1550  11EE  REM *****  INPUT                  :  DATE,PAGE,PGMID$,PGMTITL$ ****
1550  11EE  REM *****                                ****
1550  11EE  REM *****  OUTPUT                  :  PRINTER HEADING, LN.COUNT ****
1550  11EE  REM *****                                ****
1550  11EE  REM *****  RESERVED LINE          ****
1550  11EE  REM *****  NUMBERS              :  7101-7110          ****
1550  11EE  REM *****
1550  11EE
1550  11EE  7101  IF PAGE > 0 THEN PRINT CURS(12);

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.03

```

1566 11EE      LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITL$;
1573 11EE      LPRINT TAB(70);DATES
1586 11EE      PAGE=PAGE+1
158E 11EE      LPRINT "PROGRAM ";PGMID$;TAB(70);"PAGE";
15A8 11EE      LPRINT USING "####";PAGE
15B7 11EE      LPRINT
15BF 11EE      LN.COUNT=3
15C6 11EE      RETURN
15C9 11EE
15C9 11EE      8000  REM $INCLUDE: 'RACS8000.SUB' Include the DECODE SUB GROUP
15CA 11EE      REM *****
15CA 11EE      REM ****   AMBULATORY CARE DATA BASE           13 APR 85   ***
15CA 11EE      REM ****   SKIP COLE                         ***
15CA 11EE      REM ****   SUBROUTINE NAME      :   RXS8000.SUB      ***
15CA 11EE      REM ****   SCANNER PROGRAM #    :   ALL              ***
15CA 11EE      REM ****   FUNCTION              :   THIS SUBROUTINE MODULE ***
15CA 11EE      REM ****   IS A GROUPING THAT PERFORMS ***
15CA 11EE      REM ****   VARIOUS DECODING FUNCTIONS ***
15CA 11EE      REM ****   ON THE SCANNER DATA ***
15CA 11EE      REM ****
15CA 11EE      REM ****   8001  -  DECODE THE HEADER POSITIONS (POINTER 0-20) ***
15CA 11EE      REM ****   8050  -  CHECK FOR END OF JOB ***
15CA 11EE      REM ****   8100  -  PRINT THE HEADER DATA ON THE SCREEN ***
15CA 11EE      REM ****   8200  -  DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
15CA 11EE      REM ****   (RETURNED IN TEXT$ STRING VARIABLE) ***
15CA 11EE      REM ****
15CA 11EE      REM ****   INPUT              :   SHEET RECORD, RECORD LENGTH ***
15CA 11EE      REM ****
15CA 11EE      REM ****   OUTPUT              :   'TEXT$' TRING VARIABLE ***
15CA 11EE      REM ****
15CA 11EE      REM ****   RESERVED LINE ***
15CA 11EE      REM ****   NUMBERS              :   8001-8500 ***
15CA 11EE      REM *****
15CA 11EE
15CA 11EE      'DECODE THE HEADER ONLY
15CA 11EE      8001      POINTER = 0
15D1 11F0          RECORDPTR = VARPTR(SHEETREC(0))
15D8 11F2          FOR J8000 = 1 TO 21
15DF 11F2      8002      TEXT$= TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
15FD 11F2          POINTER=POINTER+1
1605 11F2          NEXT J8000
1614 11F4          PROGRAM$= LEFT$(TEXT$,3)
1623 11F8          BATCH$=  MID$(TEXT$,4,3)
1635 11FC          SERIAL$= MID$(TEXT$,7,4)
1647 1200          RUNID$=  MID$(TEXT$,11,1)
1659 1204          FORM$=   MID$(TEXT$,12,2)
1668 1208          POCKET$= MID$(TEXT$,14,1)
167D 120C          SCANERR1$=MID$(TEXT$,16,2)
168F 1210          SCANERR2$=MID$(TEXT$,18,2)
16A1 1214          SCANERR3$=MID$(TEXT$,20,2)
1683 1218          GOTO 8500
1687 1218
1687 1218      8050  REM CHECK FOR END OF JOB/END OF BATCH
1688 1218          IF PROGRAM$ = PGMID$ THEN GOTO 8500

```

Offset Data Source Line

```

16CA 1218      LPRINT STRINGS$(80,"")
16D8 1218      LPRINT
16E0 1218      LPRINT "RECORDS PROCESSED ... ";SERIAL$
16ED 1218      LPRINT "STARTED AT ..... ";B$TIMES
16FA 1218      LPRINT "ENDED AT ..... ";TIMES
1707 1218      LPRINT CHR$(12)
1712 1218      GOTO 30000
1716 1218
1716 1218      8070 REM CHECK FOR SCANNER ERRORS
1717 1218          IF POCKETS = " " GOTO 8500
1729 1218          LPRINT LITHOS;
1731 1218          LPRINT " ... SCANNER ERRORS : ";
1739 1218          LPRINT SCANERR1$;" / ";
1746 1218          LPRINT SCANERR2$;" / ";
1753 1218          LPRINT SCANERR3$
1758 1218          LN=LN+1
1763 121A      GOTO 999
1767 121A
1767 121A      8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
1768 121A          LOCATE 5,1:PRINT "PROGRAM ";PROGRAM$;
1782 121A
1782 121A          PRINT " BATCH ";BATCH$;
178F 121A          PRINT " RUN ";RUNID$;
179C 121A          PRINT " FORM ";FORM$;
17A9 121A          PRINT " POCKET ";POCKET$
17B6 121A      GOTO 8500
17BA 121A
17BA 121A      8200 REM DECODE THE RESPONSE POSITIONS
17BB 121A          POINTER = 21
17C2 121A          RECORDPTR = VARPTR(SHEETREC(0))
17C9 121A          FOR J8000 = 22 TO RECORDLENGTH
17D6 121C      8202          TEXT$ = TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
17F4 121C          POINTER=POINTER+1
17FC 121C          NEXT J8000
1800 121C
1800 121C      8500 RETURN
1810 121C
1810 121C      REM ----- END OF RXXS8000.SUB -----
1810 121C
1810 121C      9000 REM $INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
1811 121C      REM *****
1811 121C      REM ***** AMBULATORY CARE DATA BASE 13 APR 85 *****
1811 121C      REM ***** SKIP COLE *****
1811 121C      REM ***** PROGRAM NAME : RACS9000.SUB *****
1811 121C      REM ***** SCANNER PROGRAM # : ALL *****
1811 121C      REM ***** FUNCTION : THIS SUBROUTINE MODULE *****
1811 121C      REM ***** CONTROLS THE SCANNER I/O *****
1811 121C      REM *****
1811 121C      REM ***** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS *****
1811 121C      REM ***** COMMUNICATIONS MANUAL AND THE *****
1811 121C      REM ***** PRE-RELEASED SOFTWARE GUIDE *****
1811 121C      REM *****
1811 121C      REM ***** RESERVED LINE *****

```

Offset	Data	Source Line
1811	121C	REM **** NUMBERS : 9001-9100 ****
1811	121C	REM *****
1811	121C	
1811	121C	
1811	121C	REM *****
1811	121C	REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
1811	121C	REM **** ARGUMENTS: PRESET ... SEE BELOW ****
1811	121C	REM *****
1811	121C	9001 REM
1812	121C	PROTOCOL(0) = 9600 'BAUD RATE
1819	121C	PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
1820	121C	PROTOCOL(2) = 8 'DATA BITS
1827	121C	PROTOCOL(3) = 1 'STOP BITS
182E	121C	PROTOCOL(4) = 2 'RS-232 PORT
1835	121C	PROTOCOL(5) = 0 'WRITE TIME-OUT
183C	121C	PROTOCOL(6) = 0 'READ TIME-OUT
1843	121C	
1843	121C	ERRSTAT\$ = SPACES(60)
184F	121C	ARGPTR = VARPTR(PROTOCOL(0))
1856	121E	CALL SETUP (ARGPTR,ERRSTAT\$)
1867	121E	ERRMSG\$=""
1870	121E	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="SETUP ERROR "+ERRSTAT\$
188C	121E	GOTO 9100
1890	121E	
1890	121E	REM *****
1890	121E	REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
1890	121E	REM **** ARGUMENTS: CNTRLOPT ****
1890	121E	REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
1890	121E	REM **** CNTRLOPT = 2 = STOP SCANNER (S0) ****
1890	121E	REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
1890	121E	REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
1890	121E	REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
1890	121E	REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
1890	121E	REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
1890	121E	REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
1890	121E	REM *****
1890	121E	9010 REM
1891	121E	ERRSTAT\$ = SPACES(60)
1890	121E	CALL CNTRLOP (CNTRLOPT,ERRSTAT\$)
18AE	121E	ERRMSG\$=""
1887	121E	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
1803	121E	GOTO 9100
1807	121E	
1807	121E	REM *****
1807	121E	REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
1807	121E	REM ****
1807	121E	REM **** ARGUMENTS: READTYPE ****
1807	121E	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
1807	121E	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
1807	121E	REM ****
1807	121E	REM **** ARGUMENTS: RECORDLENGTH ****
1807	121E	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
1807	121E	REM **** TRANSMITTED ****
1807	121E	REM *****

Offset	Date	Source Line
1807	121E	9020 REM
1808	121E	ERRSTAT\$ = SPACE\$(60)
18E4	121E	RECORDPTR = VARPTR(SHEETREC(0))
18E8	121E	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
1904	121E	ERRMSG\$=""
1900	121E	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
192E	121E	GOTO 9100
1932	121E	
1932	121E	REM *****
1932	121E	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
1932	121E	REM ****
1932	121E	REM **** ARGUMENTS: PRINTPOS ****
1932	121E	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
1932	121E	REM **** VALUES = 0 THRU 90 ****
1932	121E	REM ****
1932	121E	REM **** ARGUMENTS: PSTRINGS ****
1932	121E	REM **** TEXT TO BE PRINTED ON THE FORM ****
1932	121E	REM ****
1932	121E	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
1932	121E	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
1932	121E	REM *****
1932	121E	9030 REM
1933	121E	ERRSTAT\$ = SPACE\$(60)
193F	121E	RECORDPTR = VARPTR(SHEETREC(0))
1946	121E	CALL TPRINT(PRINTPOS,PSTRINGS,ERRSTAT\$)
195B	1224	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
1977	1224	GOTO 9100
197B	1224	
197B	1224	9100 RETURN
197E	1224	REM -----END OF SUBROUTINE RACS9000.SUB -----
197E	1224	
197E	1224	REM END OF SUBROUTINES =====
197E	1224	
197E	1224	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
197F	1224	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1987	1224	LPRINT "ERASING FILE ";DATFIL\$
1994	1224	BEEP
1998	1224	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
19A4	1224	CLOSE
19A8	1224	OPEN DATFIL\$ FOR OUTPUT AS #1
19BA	1224	PRINT #1,STRING\$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
19CC	1224	CLOSE
19D0	1224	
19D0	1224	30000 REM
19D1	1224	CLOSE
19D5	1224	OPEN "O",1,"RACB00.BAT"
19E7	1224	PRINT #1,"RECOUP RACPTSRT.BAT"
19F2	1224	PRINT #1,"RACB10"
19FD	1224	CLOSE
1A01	1224	END
1A05	1224	
1A08	1224	

RACP200
PATIENT DESTING/DECODE PROGRAM

PAGE 21

06-08-87

07:22:23

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

22151 Bytes Available

14541 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

PAGE 1

06-C8-87

07:28:45

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

001A 0002 REM \$LINESIZE: 132
001A 0002 REM \$PAGESIZE: 66
001A 0002 REM \$TITLE: 'RACP300 '
001A 0002 REM \$SUBTITLE: 'PROVIDER DESTING/DECODE PROGRAM'
001A 0002 REM \$PAGE

```

Offset  Data  Source Line
-----
001A 0002 REM +-----+
001A 0002 REM | NAME: RXXP300          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | Date: 28 Feb 84      PROVIDER REGISTRATION PROGRAM   |
001A 0002 REM | Floyd Cole                                     |
001A 0002 REM | 8 JUN 87  CHANGE NAMES FOR NEW SYSTEM  DRB          |
001A 0002 REM +-----+
001A 0002 REM          PROVIDER OMR INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the provider OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          PROVIDER.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RXXP05, otherwise,
001A 0002 REM the program chains to program RXXP10 on exit.
001A 0002
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM Include the DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD          DIMENSION DEFINITIONS      *
001A 0002 '* Date: 28 Feb 84          Written by: Floyd Cole      *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002
001A 0002 DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 DIM ADCAT$(3) '(DECODE FOR ACTIVE DUTY CATEGORY)
001A 0002 DIM YN$(3) '(YES/NO ANSWERS 0=?, 1 = Y , 2=N , 3=X)
001A 0002 DIM ED.MSG$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002
001A 0002 REM $INCLUDE: 'RACCMN.MOD'      REM Include the COMMON AREA DEFINITION
  
```

```

Offset  Data    Source Line
001A 0002  *****
001A 0002  '*  NAME: RACCMN.MOD              COMMON AREA DEFINITION  *
001A 0002  '*  Date: 28 Feb 84              Written by: Floyd Cole  *
001A 0002  *****
001A 0002  '  COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A 0002  '  INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE,  MODE.
001A 0002  '
001A 0002  '  This program segment may be modified, but all files containing
001A 0002  '  an include for this segment must be re*compiled in order to
001A 0002  '  affect the changes made here.
001A 0002  '
001A 0002  '  *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002  COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELLS 'BASIC SCREEN COLORS
001A 0002  COMMON HEADERS$ '21 CHARACTER SCANNER HEADER INFO
001A 0002  COMMON TEXTS$ ' ' AINING CHARACTERS FROM SCANNER
001A 0002  COMMON PGMIDS$ 'PROGRAM OR FORM ID
001A 0002  COMMON MOLENGTH() 'DAYS IN THE MONTH
001A 0002  COMMON USERS()
001A 0002  '  *****END OF COMMON DEFINITION*****
001A 0002
001A 0002  REM $INCLUDE: 'RACDEF.MOD' REM Include the DEFAULT DEFINITIONS
001A 0002  *****
001A 0002  '*  NAME: RACP01.DEF              DEFAULT DEFINITIONS  *
001A 0002  '*  Date: 28 Feb 84              Written by: Floyd Cole  *
001A 0002  *****
001A 0002  '  Variables used in common that have a default value on start*up
001A 0002  '  will be held in this file. It is an included file so it cannot
001A 0002  '  be run in a stand*alone mode. In normal operation, this file
001A 0002  '  should be 'included' in the main program only (RACP10.BAS).
001A 0002  '
001A 0002  '  This program segment may be modified, but all files containing
001A 0002  '  an include for this segment must be re*compiled in order to
001A 0002  '  affect the changes made here.
001A 0002  '
001A 0002  '  *****START OF DEFAULT DEFINITION*****
001A 0002  FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
0042 0E6C  BACK = 1 'Background Color = Light Blue
0049 0E6C  BORD = 4 'BORDER = RED
0050 0E6E  HIDE = 4 'ALTERNATE COLOR = RED
0057 0E6E  EFORE= 14 'ERROR FOREGROUND DISPLAY
005E 0E6E  EBACK= 0 'ERROR BACKGROUND DISPLAY
0065 0E6E  BELLS = CHR$(7) 'Sound the bell
0071 0E6E
0071 0E6E  MOLENGTH(1) = 31 'JAN
0078 0E6E  MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
007F 0E6E  MOLENGTH(3) = 31 'MAR
0086 0E6E  MOLENGTH(4) = 30 'APR
008D 0E6E  MOLENGTH(5) = 31 'MAY
0094 0E6E  MOLENGTH(6) = 30 'JUN
009B 0E6E  MOLENGTH(7) = 31 'JUL
00A2 0E6E  MOLENGTH(8) = 31 'AUG

```

RACP300

PROVIDER DESTRING/DECODE PROGRAM

PAGE 4

06-08-87

07:28:45

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```

00A9 0E6E      MOLENGTH(9)  = 30      'SEP
00B0 0E6E      MOLENGTH(10) = 31      'OCT
00B7 0E6E      MOLENGTH(11) = 30      'NOV
00BE 0E6E      MOLENGTH(12) = 31      'DEC
00C5 0E6E
00C5 0E6E      DATEERRS(0)  = " "
00CE 0E6E      DATEERRS(1)  = "INVALID MONTH"
00D7 0E6E      DATEERRS(2)  = "INVALID DAY  "
00E0 0E6E      DATEERRS(3)  = "DAY TOO LARGE FOR MONTH CODED"
00E9 0E6E
00E9 0E6E      MAXLENGTH   = 80      'MAXIMUM LENGTH OF OUTPUT RECORD
00FD 0E7D      PADS        = " "      'PAD CHARACTER FOR SHORT RECORDS
00F9 0E74
00F9 0E74      ' *****END OF DEFAULT DEFINITION*****
00F9 0E74
00F9 0E74      KEY OFF
00FF 0E74
00FF 0E74      REM *****
00FF 0E74      REM THE FOLLOWING VARIABLES ARE UNIQUE TO EACH PROGRAM AND MUST
00FF 0E74      REM BE CHANGED.
00FF 0E74      REM *****
00FF 0E74      PGMITLS = "PROVIDER REGISTRATION"
0108 0E78
0108 0E78      PGMIDS = "300"          'VALUE RECEIVED FROM THE SCANNER
0111 0E78      'IN HEADER VARIABLE 'PROGRAMS'
0111 0E78
0111 0E78      DATFIL$ = "PROVIDER.DAT" 'FILE TO BE INPUT TO FOCUS
011A 0E7C
011A 0E7C      REM LENGTH OF STRING RECEIVED FROM THE OMR....
011A 0E7C      HEADER = 21
0121 0E7E      RESPONSE= 88
0128 0E80      RECORDLENGTH = HEADER + RESPONSE
0133 0E82
0133 0E82      REM *****
0133 0E82
0133 0E82      BTIMES=TIMES          'SCAN START TIME
013C 0E86
013C 0E86      REM ARMY ACTIVE DUTY TABLE
013C 0E86      ADCATS(0)=" "
0145 0E86      ADCATS(1)="E"
014E 0E86      ADCATS(2)="O"
0157 0E86      ADCATS(3)="W"
0160 0E86
0160 0E86      REM YES/NO TABLE
0160 0E86      YNS(0)="?" : YNS(1)="Y" : YNS(2)="N" : YNS(3)="X"
0184 0E86
0184 0E86      REM $PAGE

```

Offset	Data	Source Line
0184	0E86	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
0189	0E86	CLS
0180	0E86	GOSUB 7000 'PRINT SCREEN HEADING
0192	0E86	
0192	0E86	REM *****
0192	0E86	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
0192	0E86	REM *****
0192	0E86	REM
0192	0E86	OPEN DATFIL\$ FOR APPEND AS #1
01A4	0E86	
01A4	0E86	REM *****
01A4	0E86	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
01A4	0E86	REM *****
01A4	0E86	LPRINT CHR\$(15);
01AF	0E86	WIDTH "LPT1:",160
01B9	0E86	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
01C5	0E88	COLOR 14
01CC	0E88	LOCATE 11,26 : PRINT "PROVIDER REGISTRATION FORM"
01E1	0E88	COLOR FORE,BACK,BORD
01F7	0E88	
01F7	0E88	REM *****
01F7	0E88	REM **** COMMUNICATIONS SETUP ****
01F7	0E88	REM *****
01F7	0E88	REM PROTOCOL
01F7	0E88	GOSUB 9001
01FC	0E88	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0216	0E8C	
0216	0E8C	REM START SCANNER (S1)
0216	0E8C	CNTRLOPT =1 :GOSUB 9010
0222	0E8E	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
023C	0E8E	
023C	0E8E	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
0251	0E8E	READTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
0258	0E90	
0258	0E90	REM *****
0258	0E90	REM **** SET SCAN SHEET CALL ****
0258	0E90	REM *****
0258	0E90	REM
0258	0E90	10 REM - RETURN POINT TO READ NEXT SHEET
0259	0E90	
0259	0E90	AS=INKEY\$
0262	0E94	IF AS=CHR\$(27) THEN GOTO 25000
0278	0E94	
0278	0E94	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
0270	0E94	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
0299	0E98	
0299	0E98	TEXT\$="" 'CLEAR THE INPUT AREA
02A2	0E98	GOSUB 8000 'DECODE HEADER
02A7	0E98	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
02AC	0E98	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
02B1	0E98	LITHO\$=MID\$(TEXT\$,101,8)
02C3	0E9C	GOSUB 8070 'CHECK FOR SCANNER ERRORS
02C8	0E9C	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

Offset	Data	Source Line
02C0	0E9C	
02C0	0E9C	REM \$INCLUDE: 'RXXM301.MOD' INCLUDE THE PROVIDER REFORMAT/EDIT MOD
02C0	0E9C	*****
02C0	0E9C	REM **** AMBULATORY CARE INFORMATION SYSTEM 23 APR 85 ****
02C0	0E9C	REM **** SKIP COLE ****
02C0	0E9C	REM **** MODULE NAME : RXXM301.MOD ****
02C0	0E9C	REM **** SCANNER PROGRAM # : 300-PROVIDER REGISTRATION ****
02C0	0E9C	REM ****
02C0	0E9C	REM **** PURPOSE : DESTING/DECODE AND EDIT ****
02C0	0E9C	REM **** THE OMR STRING. ****
02C0	0E9C	REM ****
02C0	0E9C	*****
02C0	0E9C	REM **** RESERVED LINE NUMBERS 100-199 ****
02C0	0E9C	*****
02C0	0E9C	
02C0	0E9C	N.ERR =0 'COUNTS THE NUMBER OF ERRORS
02D4	0E9E	
02D4	0E9E	REM TYPE OF REGISTRATION
02D4	0E9E	100 TORS=MID\$(TEXT\$,22,1)
02E6	0EA2	
02E6	0EA2	REM *** REGISTRATION DATE ***
02E6	0EA2	105 RDATE\$ = MID\$(TEXT\$,23,6)
02F8	0EA6	CK.5000\$ = RDATE\$
0301	0EAA	CK.5010\$ = RDATE\$
030A	0EAE	GOSUB 5000 'DATE CHECK
030F	0EAE	GOSUB 5010 'NUMERIC STRING CHECK
0314	0EAE	IF RT.5000 = 0 AND RT.5010 = 0 THEN GOTO 110
0338	0EB2	N.ERR=N.ERR+1
0340	0EB2	ED.MSG\$(N.ERR)="02-04 REGISTRATION DATE "+DATEERR\$(RT.5000)
0366	0EB2	
0366	0EB2	REM *** LITHO CODE ***
0366	0EB2	110 LITHO\$ = MID\$(TEXT\$,101,8)
0378	0EB2	CK.5010\$ = LITHO\$
0381	0EB2	GOSUB 5010 'NUMERIC STRING CHECK
0386	0EB2	IF RT.5010 = 0 THEN GOTO 115
0395	0EB2	N.ERR=N.ERR+1: ED.MSG\$(N.ERR)="LITHO CODE"
0381	0EB2	
0381	0EB2	REM *** PROVIDERS NAME ***
0381	0EB2	115 PROVLNAME\$= MID\$(TEXT\$,30,16)
03C3	0EB6	PROVFNAME\$= MID\$(TEXT\$,46,13)
03D5	0EBA	PROVMINIT\$= MID\$(TEXT\$,59,1)
03E7	0EBE	
03E7	0EBE	REM *** PROVIDERS SSN ***
03E7	0EBE	120 PROVSSN\$=MID\$(TEXT\$,60,9)
03F9	0EC2	CK.5010\$ = PROVSSN\$
0402	0EC2	GOSUB 5010 'NUMERIC STRING CHECK
0407	0EC2	IF RT.5010 = 0 THEN GOTO 125
0416	0EC2	N.ERR=N.ERR+1: ED.MSG\$(N.ERR)="09 PROVIDER SSN "
0432	0EC2	
0432	0EC2	REM *** PROVIDERS ID ***
0432	0EC2	125 PROVID\$= LEFT\$(PROVLNAME\$,1) + RIGHT\$(PROVSSN\$,4)
044F	0EC6	
044F	0EC6	REM *** PROVIDERS STATUS/POSITION ***

Offset	Data	Source Line
044F	DEC6	130 PROVSTATS=MID\$(TEXT\$,69,1)
0461	OECA	PROVPOSNS=MID\$(TEXT\$,70,2)
0473	OECE	
0473	OECE	REM *** PROVIDERS SERVICE CODE *** (MAY BE BLANK)
0473	OECE	135 SERVICE=VAL(MID\$(TEXT\$,72,1))
0489	OED0	
0489	OED0	REM *** PROVIDERS CATEGORY CODE ** (MUST BE 1 - 7)
0489	OED0	140 CATEG=VAL(MID\$(TEXT\$,29,1))
049F	OED2	
049F	OED2	REM *** COMPUTE PROVIDER CODE ***
049F	OED2	145 PROVCD= (SERVICE*10) + CATEG
04AE	OED4	PROVCATS = RIGHTS(STR\$(PROVCD),2)
04C1	OED8	
04C1	OED8	REM *** DECODE/CREATE/EDIT THE COMPOSITE JOB CATEGORY, GRADE ***
04C1	OED8	
04C1	OED8	150 IF CATEG < 4 THEN GOTO 155 ELSE GOTO 170
04D4	OED8	
04D4	OED8	155 REM CATEGORY = OFF, ENL OR WARRANT (1,2, OR 3)
04D5	OED8	X=VAL(MID\$(TEXT\$,109,1))
04EB	OEDA	PAYGRADE\$=ADCAT\$(X) + MID\$(TEXT\$,73,1)
0500	OEDE	160 ON (SERVICE+1) GOTO 175, 162, 164, 166, 166, 168, 168, 168
0526	OEDE	REM ERR ARMY ARFC NAVY MRNE CG PHS OTH
0526	OEDE	
0526	OEDE	REM *** ARMY FIELDS ***
0526	OEDE	162 JOBCODE\$= MID\$(TEXT\$,74,4) 'MOS + SSI
0538	OEE2	IF CATEG = 1 OR CATEG = 2 THEN JOBCODE\$=JOBCODE\$+MID\$(TEXT\$,78,2)
0560	OEE2	IF CATEG = 3 THEN JOBCODE\$=JOBCODE\$+MID\$(TEXT\$,80,2)
0580	OEE2	JOBCODE\$ = JOBCODE\$+SPACES(1) 'PAD TO 7
059C	OEE2	GOTO 180
05A0	OEE2	
05A0	OEE2	REM *** AIR FORCE FIELDS ***
05A0	OEE2	164 JOBCODE\$= MID\$(TEXT\$,82,7)
05B2	OEE2	GOTO 180
05B6	OEE2	
05B6	OEE2	REM *** NAVY/MARINE FIELDS ***
05B6	OEE2	166 JOBCODE\$ = MID\$(TEXT\$,89,6) + SPACES(1) 'PAD TO 7
0503	OEE2	GOTO 180
05D7	OEE2	
05D7	OEE2	REM *** CG/PHS/OTHER FIELDS ***
05D7	OEE2	168 JOBCODE\$ = SPACES(7) 'PAD TO 7
05E3	OEE2	GOTO 180
05E7	OEE2	
05E7	OEE2	REM CATEGORY IS CIVILIAN, CONTRACT, CONSULANT OR OTHER
05E7	OEE2	170 PAYGRADE\$= MID\$(TEXT\$,99,2)
05F9	OEE2	JOBCODE\$= MID\$(TEXT\$,95,4) + SPACES(3) 'PAD TO 7 CHAR
0616	OEE2	GOTO 180
061A	OEE2	
061A	OEE2	REM *** ERROR PROCESSING ROUTINE ***
061A	OEE2	175 N.ERR=N.ERR+1
0622	OEE2	ED.MSG\$(N.ERR)="05 INVALID COMBINATION OF CATEGORY AND "
0636	OEE2	ED.MSG\$(N.ERR)=ED.MSG\$(N.ERR)+"MILITARY BRANCH "
064C	OEE2	GOTO 180
0650	OEE2	
0650	OEE2	180 REM MODULE EXIT POINT

Offset Data Source Line IBM Personal Computer BASIC Compiler V1.00

```

0651 OEE2 REM -----END OF MODULE RXXM301.MOD-----
0651 OEE2
0651 OEE2 IF N.ERR = 0 THEN GOTO 997
0660 OEE2 LPRINT "LITHO # ";LITHOS;" ... ERRORS"
0672 OEE2 FOR 1997 = 1 TO N.ERR
067F OEE4 LPRINT USING "### ";1997;
068B OEE6 LPRINT "=> ";ED.MSG$(1997)
06A1 OEE6 NEXT 1997
06B2 OEE6 LN.COUNT = LN.COUNT + N.ERR + 1
06B8 OEE8 CNTRLOPT = 6
06C5 OEE8 GOSUB 9010 'REJECT THE FORM
06CA OEE8 GOTO 998 'BYPASS THE DISK WRITER....
06CE OEE8
06CE OEE8 997 REM $INCLUDE 'RXXM302.MOD' REM INCLUDE THE PROVIDER DISK WRITER
06CF OEE8 REM *****
06CF OEE8 REM **** AMBULATORY CARE INFORMATION SYSTEM 26 APR 85 ****
06CF OEE8 REM **** SKIP COLE ****
06CF OEE8 REM **** MODULE NAME : RXXM302.MOD ****
06CF OEE8 REM **** SCANNER PROGRAM # : 300-PROVIDER REGISTRATION ****
06CF OEE8 REM ****
06CF OEE8 REM **** PURPOSE : CREATE AND WRITE THE DISK ****
06CF OEE8 REM **** RECORD FOR INPUT TO FOCUS ****
06CF OEE8 REM *****
06CF OEE8 REM **** OUTPUT RECORD TYPES: ****
06CF OEE8 REM **** 1 = INITIAL RECORD ****
06CF OEE8 REM **** 2 = CHANGE ****
06CF OEE8 REM **** 3 = PCS/CLOSE OUT ****
06CF OEE8 REM *****
06CF OEE8 REM **** RESERVED LINE NUMBERS : 200-299 ****
06CF OEE8 REM *****
06CF OEE8
06CF OEE8 REM BUILD THE OUTPUT RECORD
06CF OEE8
06CF OEE8 RECOUT$="300"+ TOR$ 'TRANSACTION CODE 300 + TYPE OF RECORD
06D0 OEEC
06D0 OEEC RECOUT$= RECOUT$ + PROVID$ + RDATE$ + PROVCAT$
06F6 OEEC
06F6 OEEC RECOUT$= RECOUT$ + PROVFNAMES + PROVLNAMES + PROVMINITS
070F OEEC
070F OEEC RECOUT$= RECOUT$ + PROVSSNS + PROVSTAT$ + PROVPOSNS
0728 OEEC
0728 OEEC RECOUT$= RECOUT$ + PAYGRADE$ + JOBCODE$ + LITHOS
0741 OEEC
0741 OEEC PAD=MAXLENGTH - LEN(RECOUT$) 'FIND OUT HOW SHORT THE RECORD IS
074F OEE8 RECOUT$ = RECOUT$ + STRING$(PAD,PAD$) 'PAD THE RECORD WITH FILL CHAR
0762 OEE8
0762 OEE8 PRINT #1,RECOUT$
076D OEE8
076D OEE8 REM -----END OF MODULE RXXM302.MOD-----
076D OEE8
076D OEE8
076D OEE8 998 REM CONTINUE
076E OEE8
076E OEE8 999 READTYPE = 2

```

```

Offset  Data   Source Line
0775 0EEE      IF LW.COUNT > 48 THEN GOSUB 7100  'PRINTER HEADING
0785 0EEE      GOTO 10
0789 0EEE
0789 0EEE  REM  END OF SCAN/DECODE/WRITE LOOP =====
0789 0EEE  1000  REM $INCLUDE: 'RACS1000.SUB' Include the VERIFY LOGON SUB
078A 0EEE  REM *****
078A 0EEE  REM * NAME: RACS1000          LOGON VERIFICATION SUBROUTINE      *
078A 0EEE  REM * Date: 28 Feb 84      PATIENT REGISTRATION PROGRAM        *
078A 0EEE  REM *****
078A 0EEE  REM          PATIENT OMR INPUT PROGRAM                        *
078A 0EEE  REM          *
078A 0EEE  REM This program verifies user is logged on properly. If there is no *
078A 0EEE  REM valid user logged on at the time of execution, this subroutine will*
078A 0EEE  REM chain to the logon program RACP05, otherwise a return is issued. *
078A 0EEE  REM *****
078A 0EEE  REM  RESERVED LINE NUMBERS ARE 1001 THRU 1010
078A 0EEE  REM *****
078A 0EEE  1001 OPEN "I",1,"RACLOG.DAT"
079C 0EEE      IF EOF(1) THEN 1002          'MAKE THEM LOG ON FIRST
07AA 0EEE      INPUT #1,USERS(1),DTS,TMS,PIDS
07CB 0EFA      IF USERS(1) = "" THEN 1002    'MAKE THEM LOG ON FIRST
07D9 0EFA      IF USERS(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
07E7 0EFA      CLOSE 1
07EE 0EFA      SCREEN 0,1,0,0
0804 0EFA      COLOR FORE,BACK,BORD
081A 0EFA      CLS
081E 0EFA      RETURN
0821 0EFA
0821 0EFA  1002 CLOSE
0825 0EFA      CHAIN "RACP05"
082C 0EFA  '=====END OF LOGON VERIFY SUBROUTINE 1000=====
082C 0EFA
082C 0EFA  2000  REM $INCLUDE: 'RACS2000.SUB' Include the REPLY/DELAY SUB
082D 0EFA  REM *****
082D 0EFA  REM ****  AMBULATORY CARE DATA BASE          13 APR 85      ****
082D 0EFA  REM ****                                SKIP COLE      ****
082D 0EFA  REM ****  SUBROUTINE NAME      :  RACS2000.SUB          ****
082D 0EFA  REM ****  SCANNER PROGRAM #    :  ALL                  ****
082D 0EFA  REM ****  FUNCTION              :  THIS SUBROUTINE MODULE ****
082D 0EFA  REM ****                                SERVERS AS A WAIT AND REPLY ****
082D 0EFA  REM ****                                ENTRY MODULE      ****
082D 0EFA  REM ****  INPUT                  :  SINGLE KEYBOARD ENTRY ****
082D 0EFA  REM ****                                ****
082D 0EFA  REM ****  OUTPUT                  :  KEYBOARD ENTRY - UPPER CASE ****
082D 0EFA  REM ****                                ****
082D 0EFA  REM ****  RESERVED LINE          ****
082D 0EFA  REM ****            NUMBERS      :  2001-2010          ****
082D 0EFA  REM *****
082D 0EFA  2001  REM REPLY FUNCTION
082E 0EFA  2002  REPLY$=INKEY$ : IF REPLY$="" THEN 2002
0842 0EFE      REPLY=ASC(REPLY$)
084C 0F00      IF REPLY > 90 THEN REPLY$=CHR$(REPLY XOR 32)  'CONVERT TO CAPS
0767 0F00      IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$="?"

```

```

Offset Data Source Line
0893 0F00 RETURN
0896 0F00
0896 0F00 5000 REM $INCLUDE: 'RACS5000.SUB' Include the DATE EDITOR SUB
0897 0F00 REM *****
0897 0F00 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0897 0F00 REM **** SKIP COLE ****
0897 0F00 REM **** SUBROUTINE NAME : RXXS5000.SUB ****
0897 0F00 REM **** SCANNER PROGRAM # : ALL ****
0897 0F00 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0897 0F00 REM **** PERFORMS A DATE EDIT ****
0897 0F00 REM ****
0897 0F00 REM **** INPUT : DATE TO BE CHECKED MUST BE ****
0897 0F00 REM **** IN THE VARIABLE NAMED ****
0897 0F00 REM **** 'CK.5000$' ****
0897 0F00 REM **** IN THE FORMAT "YYMMDD" ****
0897 0F00 REM ****
0897 0F00 REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
0897 0F00 REM **** VARIABLE. IF THIS VARIABLE ****
0897 0F00 REM **** CONTAINS ANY NUMBER OTHER ****
0897 0F00 REM **** THAN 0, AN ERROR WAS FOUND ****
0897 0F00 REM **** IN THE DATE. ****
0897 0F00 REM ****
0897 0F00 REM **** RESERVED LINE ****
0897 0F00 REM **** NUMBERS : 5001-5009 ****
0897 0F00 REM *****
0897 0F00 RT.5000 = 0
089E 0F00 CKYEAR = VAL(LEFT$(CK.5000$,2)) 'YEAR NUMERIC VALUE
0881 0F02 CKMONTH = VAL(MID$(CK.5000$,3,2)) 'MONTH NUMERIC VALUE
08C7 0F04 CKDAY = VAL(RIGHT$(CK.5000$,2)) 'DAY NUMERIC VALUE
08DA 0F06
08DA 0F06 IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
08FD 0F06 IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
0906 0F06 IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
091C 0F06 IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
0932 0F06
0932 0F06 REM LEAP YEAR CHECK
0932 0F06 MOLENGTH(2) = 28
0939 0F06 IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY
0948 0F06 IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR
0950 0F06 MOLENGTH(2) = 29
0964 0F06
0964 0F06 5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
0983 0F06
0983 0F06 5009 RETURN
0986 0F06
0986 0F06 REM -----END OF SUBROUTINE 5000 -----
0986 0F06
0986 0F06
0986 0F06 5010 REM $INCLUDE: 'RACS5010.SUB' Include the NUMERIC STRING EDITOR
0987 0F06 REM *****
0987 0F06 REM **** AMBULATORY CARE DATA BASE 1 MAY 85 ****
0987 0F06 REM **** SKIP COLE ****
0987 0F06 REM **** SUBROUTINE NAME : RXXS5010.SUB ****
0987 0F06 REM **** SCANNER PROGRAM # : ALL ****

```

Offset	Data	Source Line
0987	0F06	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0987	0F06	REM **** PERFORMS A NUMERIC STRING ****
0987	0F06	REM **** EDIT. ****
0987	0F06	REM ****
0987	0F06	REM **** INPUT : STRING TO BE EDITED IS IN ****
0987	0F06	REM **** THE VARIABLE NAMED ****
0987	0F06	REM **** 'CK.5010\$' ****
0987	0F06	REM ****
0987	0F06	REM **** OUTPUT : 'RT.5010' IS THE RETURN CODE ****
0987	0F06	REM **** VARIABLE. IF THIS VARIABLE ****
0987	0F06	REM **** CONTAINS ANY NUMBER OTHER ****
0987	0F06	REM **** THAN 0, AN ERROR WAS FOUND ****
0987	0F06	REM **** IN THE STRING. ****
0987	0F06	REM ****
0987	0F06	REM **** RESERVED LINE ****
0987	0F06	REM **** NUMBERS : 5011-5019 ****
0987	0F06	REM ****
0987	0F06	RT.5010 = 0
098E	0F06	
098E	0F06	FOR I5010 = 1 TO LEN(CK.5010\$)
099E	0F08	J5010= ASC(MID\$(CK.5010\$,I5010,1))
09B2	0F0C	IF J5010 < 48 OR J5010 > 57 THEN RT.5010 = RT.5010 + 1
09DA	0F0C	NEXT I5010
09EB	0F0C	
09EB	0F0C	RETURN
09EE	0F0C	REM ----- END OF SUBROUTINE 5010 -----
09EE	0F0C	
09EE	0F0C	7000 REM \$INCLUDE: 'RACS7000.SUB' Include the SCREEN HEADER SUB
09EF	0F0C	REM ****
09EF	0F0C	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
09EF	0F0C	REM **** SKIP COLE ****
09EF	0F0C	REM **** SUBROUTINE NAME : RACS7000.SUB ****
09EF	0F0C	REM **** SCANNER PROGRAM # : ALL ****
09EF	0F0C	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
09EF	0F0C	REM **** PRINTS THE STANDARD SCREEN ****
09EF	0F0C	REM **** HEADING. ****
09EF	0F0C	REM **** INPUT : COMMON VARIABLE USERS(2) ****
09EF	0F0C	REM **** SYSTEM DATE ****
09EF	0F0C	REM ****
09EF	0F0C	REM **** OUTPUT : SCREEN HEADING ****
09EF	0F0C	REM ****
09EF	0F0C	REM **** RESERVED LINE ****
09EF	0F0C	REM **** NUMBERS : 7001-7010 ****
09EF	0F0C	REM ****
09EF	0F0C	7001 LOCATE 1,1
09F9	0F0C	PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
0A01	0F0C	LOCATE 1,65
0A0E	0F0C	PRINT DATE\$;
0A16	0F0C	LOCATE 2,1
0A23	0F0C	PRINT "USER : ";USERS(1)
0A30	0F0C	RETURN
0A33	0F0C	7100 REM \$INCLUDE: 'RACS7100.SUB' Include the PRINTER HEADER SUB
0A34	0F0C	REM ****

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

0A34 OFOC REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0A34 OFOC REM **** SKIP COLE ****
0A34 OFOC REM **** SUBROUTINE NAME : RXXS7100.SUB ****
0A34 OFOC REM **** SCANNER PROGRAM # : ALL ****
0A34 OFOC REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0A34 OFOC REM **** PRINTS THE STANDARD HEADING ****
0A34 OFOC REM **** ON THE PRINTER. ****
0A34 OFOC REM **** INPUT : DATE,PAGE,PGMIDS,PGMTITLS ****
0A34 OFOC REM **** ****
0A34 OFOC REM **** OUTPUT : PRINTER HEADING, LN.COUNT ****
0A34 OFOC REM **** ****
0A34 OFOC REM **** RESERVED LINE ****
0A34 OFOC REM **** NUMBERS : 7101-7110 ****
0A34 OFOC REM ****
0A34 OFOC 7101 IF PAGE > 0 THEN LPRINT CHR$(12);
0A4A OFOC LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITLS;
0A57 OFOC LPRINT TAB(70);DATES$
0A6A OFOC PAGE=PAGE+1
0A72 OFOC LPRINT "PROGRAM ";PGMIDS;TAB(70);"PAGE";
0A8F OFOC LPRINT USING "###";PAGE
0A9B OFOC LPRINT
0AA3 OFOC LN.COUNT=3
0AAA OFOC RETURN
0AAD OFOC
0AAD OFOC 8000 REM $INCLUDE: 'RACS8000.SUB' Include the DECODE SUB GROUP
0AAE OFOC REM ****
0AAE OFOC REM **** AMBULATORY CARE DATA BASE 13 APR 85 ***
0AAE OFOC REM **** SKIP COLE ***
0AAE OFOC REM **** SUBROUTINE NAME : RXXS8000.SUB ***
0AAE OFOC REM **** SCANNER PROGRAM # : ALL ***
0AAE OFOC REM **** FUNCTION : THIS SUBROUTINE MODULE ***
0AAE OFOC REM **** IS A GROUPING THAT PERFORMS ***
0AAE OFOC REM **** VARIOUS DECODING FUNCTIONS ***
0AAE OFOC REM **** ON THE SCANNER DATA ***
0AAE OFOC REM ****
0AAE OFOC REM **** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
0AAE OFOC REM **** 8050 - CHECK FOR END OF JOB ***
0AAE OFOC REM **** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
0AAE OFOC REM **** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
0AAE OFOC REM **** (RETURNED IN TEXT$ STRING VARIABLE) ***
0AAE OFOC REM ****
0AAE OFOC REM **** INPUT : SHEET RECORD, RECORD LENGTH ***
0AAE OFOC REM ****
0AAE OFOC REM **** OUTPUT : 'TEXT$' TRING VARIABLE ***
0AAE OFOC REM ****
0AAE OFOC REM **** RESERVED LINE ***
0AAE OFOC REM **** NUMBERS : 8001-8500 ***
0AAE OFOC REM ****
0AAE OFOC
0AAE OFOC 'DECODE THE HEADER ONLY
0AAE OFOC 8001 POINTER = 0
0AB5 OFOC RECORDPTR = VARPTR(SHEETREC(0))
0ABC OF10 FOR J8000 = 1 TO 21

```

Offset Data Source Line

```

0AC3 0F10 8002      TEXTS= TEXTS+CHR$(PEEK(RECORDPTR + POINTER))
0AE1 0F10          POINTER=POINTER+1
0AE9 0F10          NEXT J8000
0AF8 0F12          PROGRAMS= LEFT$(TEXT$,3)
0B07 0F16          BATCHS=  MID$(TEXT$,4,3)
0B19 0F1A          SERIALS= MID$(TEXT$,7,4)
0B2B 0F1E          RUNIDS=  MID$(TEXT$,11,1)
0B3D 0F22          FORMS=   MID$(TEXT$,12,2)
0B4F 0F26          POCKETS= MID$(TEXT$,14,1)
0B61 0F2A          SCANERR1$=MID$(TEXT$,16,2)
0B73 0F2E          SCANERR2$=MID$(TEXT$,18,2)
0B85 0F32          SCANERR3$=MID$(TEXT$,20,2)
0B97 0F36          GOTO 8500
0B9B 0F36
0B9B 0F36 8050 REM CHECK FOR END OF JOB/END OF BATCH
0B9C 0F36          IF PROGRAMS = PGMIDS THEN GOTO 8500
0BAE 0F36          LPRINT STRING$(80,"")
0BBC 0F36          LPRINT
0BC4 0F36          LPRINT "RECORDS PROCESSED ... ";SERIALS
0BD1 0F36          LPRINT "STARTED AT ..... ";BTIMES
0BDE 0F36          LPRINT "ENDED   AT ..... ";TIMES
0BEB 0F36          LPRINT CHR$(12)
0BF6 0F36          GOTO 30000
0BFA 0F36
0BFA 0F36 8070 REM CHECK FOR SCANNER ERRORS
0BFB 0F36          IF POCKETS = " " GOTO 8500
0C0D 0F36          LPRINT LITHOS;
0C15 0F36          LPRINT " ... SCANNER ERRORS : ";
0C1D 0F36          LPRINT SCANERR1$;" / ";
0C2A 0F36          LPRINT SCANERR2$;" / ";
0C37 0F36          LPRINT SCANERR3$
0C3F 0F36          LN=LN+1
0C47 0F38          GOTO 999
0C4B 0F38
0C4B 0F38 8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
0C4C 0F38          LOCATE 5,1:PRINT "PROGRAM ";PROGRAMS;
0C66 0F38
0C66 0F38          PRINT " BATCH ";BATCHS;
0C73 0F38          PRINT " RUN ";RUNIDS;
0C80 0F38          PRINT " FORM ";FORMS;
0C8D 0F38          PRINT " POCKET ";POCKETS
0C9A 0F38          GOTO 8500
0C9E 0F38
0C9E 0F38 8200 REM DECODE THE RESPONSE POSITIONS
0C9F 0F38          POINTER = 21
0CA6 0F38          RECORDPTR = VARPTR(SHEETREC(0))
0CAD 0F38          FOR J8000 = 22 TO RECORDLENGTH
0CBA 0F3A 8202      TEXTS = TEXTS+CHR$(PEEK(RECORDPTR + POINTER))
0CDB 0F3A          POINTER=POINTER+1
0CED 0F3A          NEXT J8000
0CF1 0F3A
0CF1 0F3A 8500 RETURN
0CF4 0F3A
0CF4 0F3A REM ----- END OF RXXS8000.SUB -----

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

OCF4 0F3A
OCF4 0F3A 9000 REM $INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
OCF5 0F3A REM *****
OCF5 0F3A REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
OCF5 0F3A REM **** SKIP COLE ****
OCF5 0F3A REM **** PROGRAM NAME : RACS9000.SUB ****
OCF5 0F3A REM **** SCANNER PROGRAM # : ALL ****
OCF5 0F3A REM **** FUNCTION : THIS SUBROUTINE MODULE ****
OCF5 0F3A REM **** CONTROLS THE SCANNER I/O ****
OCF5 0F3A REM ****
OCF5 0F3A REM **** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS ****
OCF5 0F3A REM **** COMMUNICATIONS MANUAL AND THE ****
OCF5 0F3A REM **** PRE-RELEASED SOFTWARE GUIDE ****
OCF5 0F3A REM ****
OCF5 0F3A REM *****
OCF5 0F3A REM **** RESERVED LINE ****
OCF5 0F3A REM **** NUMBERS : 9001-9100 ****
OCF5 0F3A REM *****
OCF5 0F3A
OCF5 0F3A
OCF5 0F3A REM *****
OCF5 0F3A REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
OCF5 0F3A REM **** ARGUMENTS: PRESET ... SEE BELOW ****
OCF5 0F3A REM *****
OCF5 0F3A 9001 REM
OCF6 0F3A PROTOCOL(0) = 9600 'BAUD RATE
OCFD 0F3A PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
0004 0F3A PROTOCOL(2) = 8 'DATA BITS
0008 0F3A PROTOCOL(3) = 1 'STOP BITS
0012 0F3A PROTOCOL(4) = 2 'RS-232 PORT
0019 0F3A PROTOCOL(5) = 0 'WRITE TIME-OUT
0020 0F3A PROTOCOL(6) = 0 'READ TIME-OUT
0027 0F3A
0027 0F3A ERRSTAT$ = SPACES(60)
0033 0F3A ARGPTR = VARPTR(PROTOCOL(0))
003A 0F3C CALL SETUP (ARGPTR,ERRSTAT$)
0048 0F3C ERRMSG$=""
0054 0F3C IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="SETUP ERROR "+ERRSTAT$
0070 0F3C GOTO 9100
0074 0F3C
0074 0F3C REM *****
0074 0F3C REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
0074 0F3C REM **** ARGUMENTS: CNTRLOPT ****
0074 0F3C REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
0074 0F3C REM **** CNTRLOPT = 2 = STOP SCANNER (S0) ****
0074 0F3C REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
0074 0F3C REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
0074 0F3C REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
0074 0F3C REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
0074 0F3C REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
0074 0F3C REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
0074 0F3C REM *****
0074 0F3C 9010 REM
0075 0F3C ERRSTAT$ = SPACES(60)

```

Offset	Data	Source Line
0081	0F3C	CALL CNTRL (CNTRLOPT,ERRSTAT\$)
0092	0F3C	ERRMSG\$=""
0098	0F3C	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
00B7	0F3C	GOTO 9100
00B8	0F3C	
00B8	0F3C	REM *****
00B8	0F3C	REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
00B8	0F3C	REM ****
00B8	0F3C	REM **** ARGUMENTS: READTYPE ****
00B8	0F3C	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
00B8	0F3C	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
00B8	0F3C	REM ****
00B8	0F3C	REM **** ARGUMENTS: RECORDLENGTH ****
00B8	0F3C	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
00B8	0F3C	REM **** TRANSMITTED ****
00B8	0F3C	REM *****
00B8	0F3C	9020 REM
00B8	0F3C	ERRSTAT\$ = SPACE\$(60)
00C8	0F3C	RECORDPTR = VARPTR(SHEETREC(0))
00CF	0F3C	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
00E8	0F3C	ERRMSG\$=""
00F1	0F3C	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
0E12	0F3C	GOTO 9100
0E16	0F3C	
0E16	0F3C	REM *****
0E16	0F3C	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
0E16	0F3C	REM ****
0E16	0F3C	REM **** ARGUMENTS: PRINTPOS ****
0E16	0F3C	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
0E16	0F3C	REM **** VALUES = 0 THRU 90 ****
0E16	0F3C	REM ****
0E16	0F3C	REM **** ARGUMENTS: PSTRING\$ ****
0E16	0F3C	REM **** TEXT TO BE PRINTED ON THE FORM ****
0E16	0F3C	REM ****
0E16	0F3C	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
0E16	0F3C	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
0E16	0F3C	REM *****
0E16	0F3C	9030 REM
0E17	0F3C	ERRSTAT\$ = SPACE\$(60)
0E23	0F3C	RECORDPTR = VARPTR(SHEETREC(0))
0E2A	0F3C	CALL TPRINT(PRINTPOS,PSTRING\$,ERRSTAT\$)
0E3F	0F42	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
0E58	0F42	GOTO 9100
0E5F	0F42	
0E5F	0F42	9100 RETURN
0E62	0F42	REM -----END OF SUBROUTINE RACS9020.SUB -----
0E62	0F42	
0E62	0F42	REM END OF SUBROUTINES =====
0E62	0F42	
0E62	0F42	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
0E63	0F42	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
0E68	0F42	LPRINT "ERASING FILE ";DATFIL\$
0E78	0F42	BEEP

RACP300
PROVIDER DESTSTRING/DECODE PROGRAM

PAGE 16

06-09-87

07:28:45

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

0E7C	0F42	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
0E88	0F42	CLOSE
0E8C	0F42	OPEN DATFIL\$ FOR OUTPUT AS #1
0E9E	0F42	PRINT #1,STRING\$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
0EB0	0F42	CLOSE
0EB4	0F42	
0EB4	0F42	30000 REM
0EB5	0F42	CLOSE
0EB9	0F42	OPEN "O",1,"RACB00.BAT"
0ECB	0F42	PRINT #1,"RECOUP RACPRSRT.BAT"
0ED6	0F42	PRINT #1,"RACB10"
0EE1	0F42	CLOSE
0EE5	0F42	END
0EE9	0F42	
0EEC	0F42	

22151 Bytes Available

18186 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

PAGE 1

07-06-87

14:51:36

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

001A 0002 REM \$LINESIZE: 132
001A 0002 REM \$PAGESIZE: 66
001A 0002 REM \$TITLE: 'RACPGEN '
001A 0002 REM \$SUBTITLE: 'GENERAL FORM DESTING/DECODE PROGRAM'
001A 0002 REM \$PAGE

RACPGEN
GENERAL FORM DESTING/DECODE PROGRAM

PAGE 2
07-C6-87
14:51:36

IBM Personal Computer BASIC Compiler V1.00

```
Offset Data Source Line
001A 0002 REM +-----+
001A 0002 REM | NAME: RACPGEN          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | DATE: 27 FEB 87      GENERAL FORM PROGRAM          |
001A 0002 REM | D R BOLLING                               |
001A 0002 REM +-----+
001A 0002 REM          GENERAL FORM OMR INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the base form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RACP05, otherwise,
001A 0002 REM the program chains to program RACP10 on exit.
001A 0002
001A 0002 REM $INCLUDE: 'RACDIM.MOD' REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD          DIMENSION DEFINITIONS *
001A 0002 '* Date: 28 Feb 84          Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USER$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002
001A 0002 DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 DIM NRESP(80) '(NUMBER OF RESPONSES FOR EACH FORM)
001A 0002 DIM PROCN(80) '(NUMBER OF PROCEDURES ON EACH FORM)
001A 0002 DIM NSDX(80) '(NUMBER OF SECONDARY DX CODES ON EACH FORM)
001A 0002 DIM NOTH(80) '(NUMBER OF OTHER CODES ON EACH FORM)
001A 0002 DIM CC1(80) 'NUMBER OF PRIM DX IN COL 1
001A 0002 DIM CC2(80) 'NUMBER OF PRIM DX IN COL 2
001A 0002 DIM CC3(80) 'NUMBER OF PRIM DX IN COL 3
```

Offset	Data	Source Line
001A	0002	DIM CC4(80) 'NUMBER OF PRIM DX IN COL 4
001A	0002	DIM CLIDEFS(80) 'CLINIC DEFAULT CODES
001A	0002	DIM CLIBUBS(80) 'SINGLE BUBBLE CLINIC CODES
001A	0002	DIM REFCOS(26) 'SINGLE BUBBLE REFERRAL CODES PHYS THER
001A	0002	DIM REFCOS(26) 'SINGLE BUBBLE REFERRAL CODES OCCU THER
001A	0002	DIM OPTOMS(21) 'OPTOMOTRY OTHER CODES
001A	0002	DIM ORTHOS(107) 'ORTHO APPL OTHER CODES
001A	0002	DIM RHEUMS(11) 'RHEUMATOLOGY OTHER CODES
001A	0002	DIM SOCWOS(1) 'SOCIAL WORK OTHER CODES
001A	0002	DIM DIALYS(24) 'DIALYSIS OTHER CODES
001A	0002	DIM NEUROS(29) 'NEUROSURGERY OTHER CODES
001A	0002	DIM PHYTHS(19) 'PHYSICAL THERAPY OTHER CODES
001A	0002	DIM PSCOLS(34) 'PSYCHOLOGY OTHER CODES
001A	0002	DIM PSYCHS(30) 'PSYCHIATRY OTHER CODES
001A	0002	DIM PSYASS(4) 'PSYCHOMETRIC ASSESSMENTS CODES
001A	0002	DIM YNS(3) '(YES/NO ANSWERS 0=?, 1 = Y , 2=N , 3=X)
001A	0002	DIM ED.MSGS(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A	0002	DIM CLINIC1.PFX(5) '(PREFIX -B D F G S- FOR CLINIC #1)
001A	0002	DIM CLINIC2.PFX(6) '(PREFIX -A B C D F S- FOR CLINIC #2)
001A	0002	DIM PROVIDER.TIMES(22) '(TIME TABLE FOR PROVIDERS)
001A	0002	DIM PROCES(125) '(PROCEDURE TABLE FOR BASE FORM)
001A	0002	DIM DIAGN.TAB(225) '(DIAGNOSIS TABLE FOR BASE FORM)
001A	0002	DIM HOLDS(99) '(HOLD AREA FOR SUBROUTINE 6000)
001A	0002	DIM SPECL(09) '(SPECIAL PROGRAMS)
001A	0002	REM \$INCLUDE: 'RACCMN.MOD' REM INCLUDE THE COMMON AREA DEFINITION
001A	0002	*****
001A	0002	* NAME: RACCMN.MOD COMMON AREA DEFINITION *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A	0002	' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	' *****START OF COMMON DEFINITIONS*****
001A	0002	COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELLS 'BASIC SCREEN COLORS
001A	0002	COMMON HEADERS '21 CHARACTER SCANNER HEADER INFO
001A	0002	COMMON TEXTS ' ' AINING CHARACTERS FROM SCANNER
001A	0002	COMMON PGMIDS 'PROGRAM OR FORM ID
001A	0002	COMMON MOLENGTH() 'DAYS IN THE MONTH
001A	0002	COMMON USERS()
001A	0002	' *****END OF COMMON DEFINITION*****
001A	0002	REM \$INCLUDE: 'RACDEF.MOD' REM INCLUDE THE DEFAULT DEFINITIONS
001A	0002	*****
001A	0002	* NAME: RACP01.DEF DEFAULT DEFINITIONS *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' Variables used in common that have a default value on start*up

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

001A 0002 ' will be held in this file. It is an included file so it cannot
001A 0002 ' be run in a stand*alone mode. In normal operation, this file
001A 0002 ' should be 'included' in the main program only (RACP10.BAS).
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re*compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 '
001A 0002 ' *****START OF DEFAULT DEFINITION*****
001A 0002 FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
005E 231C BACK = 1 'Background Color = Light Blue
0065 231C BORD = 4 'BORDER = RED
006C 231E HIDE = 4 'ALTERNATE COLOR = RED
0073 231E EFORE= 14 'ERROR FOREGROUND DISPLAY
007A 231E EBACK= 0 'ERROR BACKGROUND DISPLAY
0081 231E BELL$ = CHR$(7) 'Sound the bell
008D 231E
008D 231E MOLENGTH(1) = 31 'JAN
0094 231E MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
0098 231E MOLENGTH(3) = 31 'MAR
00A2 231E MOLENGTH(4) = 30 'APR
00A9 231E MOLENGTH(5) = 31 'MAY
0080 231E MOLENGTH(6) = 30 'JUN
0087 231E MOLENGTH(7) = 31 'JUL
008E 231E MOLENGTH(8) = 31 'AUG
00C5 231E MOLENGTH(9) = 30 'SEP
00CC 231E MOLENGTH(10) = 31 'OCT
00D3 231E MOLENGTH(11) = 30 'NOV
00DA 231E MOLENGTH(12) = 31 'DEC
00E1 231E
00E1 231E DATEERR$(0) = " "
00EA 231E DATEERR$(1) = "INVALID MONTH"
00F3 231E DATEERR$(2) = "INVALID DAY "
00FC 231E DATEERR$(3) = "DAY TOO LARGE FOR MONTH CODED"
0105 231E
0105 231E MAXLENGTH = 80 'MAXIMUM LENGTH OF OUTPUT RECORD
010C 2320 PAD$ = "." 'PAD CHARACTER FOR SHORT RECORDS
0115 2324
0115 2324 ' *****END OF DEFAULT DEFINITION*****
0115 2324
0115 2324
0115 2324 KEY OFF
0118 2324
0118 2324 REM *****
0118 2324 REM PROGRAM PARAMETERS
0118 2324 NRESP(40)=282 : PROCN(40)=028 : NSDX(40)=-02 : NOTH(40)=00
0137 2324 NRESP(41)=255 : PROCN(41)=043 : NSDX(41)=-01 : NOTH(41)=00
0153 2324 NRESP(42)=270 : PROCN(42)=049 : NSDX(42)=-01 : NOTH(42)=00
016F 2324 NRESP(43)=302 : PROCN(43)=029 : NSDX(43)=-01 : NOTH(43)=00
0188 2324 NRESP(44)=255 : PROCN(44)=015 : NSDX(44)=-00 : NOTH(44)=00
01A7 2324 NRESP(45)=445 : PROCN(45)=092 : NSDX(45)=-01 : NOTH(45)=00
01C3 2324 NRESP(46)=184 : PROCN(46)=010 : NSDX(46)=-01 : NOTH(46)=00
01DF 2324 NRESP(47)=284 : PROCN(47)=041 : NSDX(47)=-01 : NOTH(47)=00

```

Offset Data Source Line

```

01FB 2324 NRESP(48)=294 : PROCN(48)=016 : NSDX(48)=-02 : NOTH(48)=00
0217 2324 NRESP(49)=272 : PROCN(49)=029 : NSDX(49)=-01 : NOTH(49)=00
0233 2324 NRESP(50)=301 : PROCN(50)=036 : NSDX(50)=-01 : NOTH(50)=00
024F 2324 NRESP(51)=385 : PROCN(51)=058 : NSDX(51)=-02 : NOTH(51)=00
0268 2324 NRESP(52)=209 : PROCN(52)=017 : NSDX(52)=-01 : NOTH(52)=00
0287 2324 NRESP(53)=209 : PROCN(53)=013 : NSDX(53)=-02 : NOTH(53)=00
02A3 2324 NRESP(54)=359 : PROCN(54)=064 : NSDX(54)=-01 : NOTH(54)=24
028F 2324 NRESP(55)=350 : PROCN(55)=017 : NSDX(55)=-01 : NOTH(55)=00
02DB 2324 NRESP(56)=372 : PROCN(56)=020 : NSDX(56)=-02 : NOTH(56)=29
02F7 2324 NRESP(57)=000 : PROCN(57)=000 : NSDX(57)=000 : NOTH(57)=00
0313 2324 NRESP(58)=289 : PROCN(58)=038 : NSDX(58)=-02 : NOTH(58)=00
032F 2324 NRESP(59)=286 : PROCN(59)=044 : NSDX(59)=-02 : NOTH(59)=00
034B 2324 NRESP(60)=300 : PROCN(60)=024 : NSDX(60)=-01 : NOTH(60)=00
0367 2324 NRESP(61)=293 : PROCN(61)=041 : NSDX(61)=-02 : NOTH(61)=00
0383 2324 NRESP(62)=332 : PROCN(62)=051 : NSDX(62)=-02 : NOTH(62)=21
039F 2324 NRESP(63)=313 : PROCN(63)=041 : NSDX(63)= 00 : NOTH(63)=107
03BB 2324 NRESP(64)=288 : PROCN(64)=025 : NSDX(64)=-01 : NOTH(64)=00
0307 2324 NRESP(65)=448 : PROCN(65)=120 : NSDX(65)=-02 : NOTH(65)=00
03F3 2324 NRESP(66)=304 : PROCN(66)=050 : NSDX(66)=-01 : NOTH(66)=00
040F 2324 NRESP(67)=315 : PROCN(67)=040 : NSDX(67)=-02 : NOTH(67)=00
042B 2324 NRESP(68)=352 : PROCN(68)=065 : NSDX(68)=-02 : NOTH(68)=19
0447 2324 NRESP(69)=249 : PROCN(69)=022 : NSDX(69)=-01 : NOTH(69)=00
0463 2324 NRESP(70)=303 : PROCN(70)=051 : NSDX(70)=-01 : NOTH(70)=00
047F 2324 NRESP(71)=332 : PROCN(71)=056 : NSDX(71)=-02 : NOTH(71)=00
049B 2324 NRESP(72)=233 : PROCN(72)=012 : NSDX(72)=-02 : NOTH(72)=00
04B7 2324 NRESP(73)=341 : PROCN(73)=010 : NSDX(73)=-02 : NOTH(73)=30
04D3 2324 NRESP(74)=375 : PROCN(74)=038 : NSDX(74)=-00 : NOTH(74)=34
04EF 2324 NRESP(75)=264 : PROCN(75)=036 : NSDX(75)=-01 : NOTH(75)=00
050B 2324 NRESP(76)=000 : PROCN(76)=000 : NSDX(76)=000 : NOTH(76)=00
0527 2324 NRESP(77)=263 : PROCN(77)=009 : NSDX(77)=-01 : NOTH(77)=11
0543 2324 NRESP(78)=296 : PROCN(78)=034 : NSDX(78)=-01 : NOTH(78)=01
055F 2324 NRESP(79)=303 : PROCN(79)=056 : NSDX(79)=-01 : NOTH(79)=00
057B 2324
057B 2324 CC1(40)=34 : CC2(40)=35 : CC3(40)=30 : CC4(40)=24
0597 2324 CC1(41)=17 : CC2(41)=19 : CC3(41)=31 : CC4(41)=00
05B3 2324 CC1(42)=31 : CC2(42)=28 : CC3(42)=11 : CC4(42)=00
05CF 2324 CC1(43)=43 : CC2(43)=34 : CC3(43)=33 : CC4(43)=30
05EB 2324 CC1(44)=37 : CC2(44)=28 : CC3(44)=28 : CC4(44)=27
0607 2324 CC1(45)=42 : CC2(45)=43 : CC3(45)=44 : CC4(45)=28
0623 2324 CC1(46)=18 : CC2(46)=18 : CC3(46)=26 : CC4(46)=00
063F 2324 CC1(47)=25 : CC2(47)=37 : CC3(47)=20 : CC4(47)=16
065B 2324 CC1(48)=48 : CC2(48)=41 : CC3(48)=42 : CC4(48)=28
0677 2324 CC1(49)=33 : CC2(49)=40 : CC3(49)=39 : CC4(49)=00
0693 2324 CC1(50)=26 : CC2(50)=35 : CC3(50)=42 : CC4(50)=22
06AF 2324 CC1(51)=44 : CC2(51)=44 : CC3(51)=46 : CC4(51)=32
06CB 2324 CC1(52)=30 : CC2(52)=24 : CC3(52)=19 : CC4(52)=00
06E7 2324 CC1(53)=25 : CC2(53)=27 : CC3(53)=18 : CC4(53)=10
0703 2324 CC1(54)=31 : CC2(54)=37 : CC3(54)=18 : CC4(54)=17
071F 2324 CC1(55)=59 : CC2(55)=62 : CC3(55)=50 : CC4(55)=41
073B 2324 CC1(56)=60 : CC2(56)=62 : CC3(56)=61 : CC4(56)=17
^757 2324 CC1(57)=00 : CC2(57)=00 : CC3(57)=00 : CC4(57)=00
0773 2324 CC1(58)=42 : CC2(58)=38 : CC3(58)=32 : CC4(58)=00
078F 2324 CC1(59)=25 : CC2(59)=24 : CC3(59)=34 : CC4(59)=12
07AB 2324 CC1(60)=40 : CC2(60)=35 : CC3(60)=39 : CC4(60)=34

```

Offset Data Source Line

```

07C7 2324      CC1(61)=30 : CC2(61)=28 : CC3(61)=32 : CC4(61)=18
07E3 2324      CC1(62)=34 : CC2(62)=30 : CC3(62)=26 : CC4(62)=16
07FF 2324      CC1(63)=12 : CC2(63)=11 : CC3(63)=00 : CC4(63)=00
081B 2324      CC1(64)=35 : CC2(64)=43 : CC3(64)=33 : CC4(64)=23
0837 2324      CC1(65)=28 : CC2(65)=31 : CC3(65)=31 : CC4(65)=15
0853 2324      CC1(66)=38 : CC2(66)=43 : CC3(66)=21 : CC4(66)=00
086F 2324      CC1(67)=25 : CC2(67)=45 : CC3(67)=37 : CC4(67)=25
088B 2324      CC1(68)=37 : CC2(68)=34 : CC3(68)=17 : CC4(68)=12
08A7 2324      CC1(69)=22 : CC2(69)=27 : CC3(69)=26 : CC4(69)=26
08C3 2324      CC1(70)=18 : CC2(70)=42 : CC3(70)=39 : CC4(70)=00
08DF 2324      CC1(71)=32 : CC2(71)=31 : CC3(71)=34 : CC4(71)=20
08FB 2324      CC1(72)=34 : CC2(72)=29 : CC3(72)=28 : CC4(72)=15
0917 2324      CC1(73)=60 : CC2(73)=52 : CC3(73)=55 : CC4(73)=14
0933 2324      CC1(74)=52 : CC2(74)=40 : CC3(74)=59 : CC4(74)=00
094F 2324      CC1(75)=40 : CC2(75)=25 : CC3(75)=25 : CC4(75)=00
096B 2324      CC1(76)=00 : CC2(76)=00 : CC3(76)=00 : CC4(76)=00
0987 2324      CC1(77)=36 : CC2(77)=32 : CC3(77)=34 : CC4(77)=28
09A3 2324      CC1(78)=30 : CC2(78)=35 : CC3(78)=29 : CC4(78)=22
09BF 2324      CC1(79)=19 : CC2(79)=23 : CC3(79)=33 : CC4(79)=12
09DB 2324
09DB 2324  REM  *** DEFAULT CLINIC CODE TABLE ***
09DB 2324      CLIDEFS(40)="BDBA" : CLIDEFS(54)="BA IA" : CLIDEFS(68)="DHDA"
09F6 2324      CLIDEFS(41)="BABA" : CLIDEFS(55)="BA JA" : CLIDEFS(69)="BBGA"
0A11 2324      CLIDEFS(42)="BHDA" : CLIDEFS(56)="BBCA" : CLIDEFS(70)="BEFA"
0A2C 2324      CLIDEFS(43)="BACA" : CLIDEFS(57)=" " : CLIDEFS(71)="FBBA"
0A47 2324      CLIDEFS(44)="BBBA" : CLIDEFS(58)="BALA" : CLIDEFS(72)="BHAA"
0A62 2324      CLIDEFS(45)="BAPA" : CLIDEFS(59)="BCCA" : CLIDEFS(73)="BFAA"
0A7D 2324      CLIDEFS(46)="BAFA" : CLIDEFS(60)="BAMA" : CLIDEFS(74)="BFBA"
0A98 2324      CLIDEFS(47)="BBFA" : CLIDEFS(61)="BBDA" : CLIDEFS(75)="BANA"
0AB3 2324      CLIDEFS(48)="BGYA" : CLIDEFS(62)="BHCA" : CLIDEFS(76)=" "
0ACE 2324      CLIDEFS(49)="BAGA" : CLIDEFS(63)="BEEA" : CLIDEFS(77)="BAQA"
0AE9 2324      CLIDEFS(50)="BBAA" : CLIDEFS(64)="BEAA" : CLIDEFS(78)="BFEA"
0B04 2324      CLIDEFS(51)="BCBA" : CLIDEFS(65)="DHBA" : CLIDEFS(79)="BBIA"
0B1F 2324      CLIDEFS(52)="BAQA" : CLIDEFS(66)="BBAB"
0B31 2324      CLIDEFS(53)="BAAA" : CLIDEFS(67)="BDAA"
0B43 2324
0B43 2324  REM  *** CLINIC CODE BY SINGLE BUBBLE ***
0B43 2324      CLIBUBS(40)="BDAABGYNBGTAJAA" 'ADOLESCENT MEDICINE
0B4C 2324      CLIBUBS(42)="BHEA" 'AUDIOLOGY/SPEECH
0B55 2324      CLIBUBS(43)="DDAA" 'CARDIOLOGY
0B5E 2324      CLIBUBS(45)="DHCA" 'DERMATOLOGY
0B67 2324      CLIBUBS(48)="BGYN" 'FAMILY PRACTICE
0B70 2324      CLIBUBS(49)="BBHA" 'GASTROENTEROLOGY
0B79 2324      CLIBUBS(51)="BGYABCAABGYN" 'GYNECOLOGY
0B82 2324      CLIBUBS(53)="BAEABAIA" 'INTERNAL MEDICINE
0B8B 2324      CLIBUBS(54)="DGBA" 'NEPHROLOGY
0B94 2324      CLIBUBS(55)="BAAA" 'NEUROLOGY
0B9D 2324      CLIBUBS(59)="BGYABCCCBGYN" 'OBSTETRICS
0BA6 2324      CLIBUBS(60)="BAMBBAHABAAA" 'ONCOLOGY
0BAF 2324      CLIBUBS(62)="BHCHBHCI" 'OPTOMETRY
0BB8 2324      CLIBUBS(63)="BEBA" 'ORTHO APPL
0BC1 2324      CLIBUBS(64)="BEBABEFA" 'ORTHOPEDICS
0BCA 2324      CLIBUBS(66)="DHCA" 'PAIN/PHYS MED
0BD3 2324      CLIBUBS(67)="BDBABGYABHABDCABGYNBIYA" 'PEDIATRICS

```

Offset Data Source Line

```

08DC 2324 CLIBUS$(68)="BEDABEDADHDA" 'PHYS THER -LAST TWO ARE TOGETHER
08E5 2324 CLIBUS$(71)="FBAABAQA" 'PREVENTIVE MEDICINE
08EE 2324 CLIBUS$(72)="BHBABJYA" 'PRIMARY CARE
08F7 2324 CLIBUS$(73)="BFDABFCA" 'PSYCHIATRY
0C00 2324 CLIBUS$(74)="BFDABFAABFCA" 'PSYCHOLOGY
0C09 2324 CLIBUS$(75)="DDADHAA" 'PULMONARY
0C12 2324 CLIBUS$(77)="BAAA" 'UROLOGY
0C18 2324 CLIBUS$(78)="BFDABFEB" 'SOCIAL WORK
0C24 2324
0C24 2324 REM *** SINGLE BUBBLE REFERRAL CODES (FORM 680-PHYS THER ONLY) ***
0C24 2324 REFCOS(00)=" " : REFCOS(09)="BBCA" : REFCOS(18)="BHAG"
0C3F 2324 REFCOS(01)="AAAA" : REFCOS(10)="BEAA" : REFCOS(19)="BHAI"
0C5A 2324 REFCOS(02)="ABAA" : REFCOS(11)="BEDA" : REFCOS(20)="BHA1"
0C75 2324 REFCOS(03)="AEAA" : REFCOS(12)="BEFA" : REFCOS(21)="BHAK"
0C90 2324 REFCOS(04)="BAAA" : REFCOS(13)="BDAA" : REFCOS(22)="BHAL"
0CAB 2324 REFCOS(05)="BACA" : REFCOS(14)="BGYA" : REFCOS(23)="BHAM"
0C06 2324 REFCOS(06)="BAKA" : REFCOS(15)="BHAA" : REFCOS(24)="BIYA"
0CE1 2324 REFCOS(07)="BAOA" : REFCOS(16)="BHAE" : REFCOS(25)="BJYA"
0CFC 2324 REFCOS(08)="BBAA" : REFCOS(17)="BHAF" : REFCOS(26)="DHCA"
0D17 2324
0D17 2324 REM *** SINGLE BUBBLE REFERRAL CODES (FORM 650-OCU THER ONLY) ***
0D17 2324 REFCOS(00)=" " : REFCOS(09)="BAAA" : REFCOS(18)="BGYA"
0D32 2324 REFCOS(01)="AAAA" : REFCOS(10)="BACA" : REFCOS(19)="BHAA"
0D4D 2324 REFCOS(02)="AAAF" : REFCOS(11)="BAKA" : REFCOS(20)="BHAB"
0D68 2324 REFCOS(03)="AABA" : REFCOS(12)="BBAA" : REFCOS(21)="BHAE"
0D83 2324 REFCOS(04)="ABAA" : REFCOS(13)="BBCA" : REFCOS(22)="BHAF"
0D9E 2324 REFCOS(05)="ABDA" : REFCOS(14)="BBDA" : REFCOS(23)="BHAG"
0DB9 2324 REFCOS(06)="AEAA" : REFCOS(15)="BDAA" : REFCOS(24)="BHEA"
0DD4 2324 REFCOS(07)="AFYA" : REFCOS(16)="BDZA" : REFCOS(25)="BIYA"
0DEF 2324 REFCOS(08)="AFYC" : REFCOS(17)="BEAA" : REFCOS(26)="DHDA"
0E0A 2324
0E0A 2324 REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0E0A 2324 REM CLINIC #1 CLINIC #2
0E0A 2324 CLINIC1.PFX$(0)=" " : CLINIC2.PFX$(0)=" "
0E1C 2324 CLINIC1.PFX$(1)="B" : CLINIC2.PFX$(1)="A"
0E2E 2324 CLINIC1.PFX$(2)="D" : CLINIC2.PFX$(2)="B"
0E40 2324 CLINIC1.PFX$(3)="F" : CLINIC2.PFX$(3)="C"
0E52 2324 CLINIC1.PFX$(4)="G" : CLINIC2.PFX$(4)="D"
0E64 2324 CLINIC1.PFX$(5)="S" : CLINIC2.PFX$(5)="F"
0E76 2324 CLINIC2.PFX$(6)="S"
0E7F 2324
0E7F 2324 REM *** ENCOUNTER FORM PROVIDER TIME TABLE ***
0E7F 2324
0E7F 2324 PROVIDER.TIME$(00)="000" ' NO TIME
0E88 2324 PROVIDER.TIME$(01)="002" ' 2 MINUTES
0E91 2324 PROVIDER.TIME$(02)="005" ' 5 MINUTES
0E9A 2324 PROVIDER.TIME$(03)="010" ' 10 MINUTES
0EA3 2324 PROVIDER.TIME$(04)="015" ' 15 MINUTES
0EAC 2324 PROVIDER.TIME$(05)="020" ' 20 MINUTES
0EB5 2324 PROVIDER.TIME$(06)="030" ' 30 MINUTES
0EBE 2324 PROVIDER.TIME$(07)="045" ' 45 MINUTES
0EC7 2324 PROVIDER.TIME$(08)="060" ' 1 HOUR
0ED0 2324 PROVIDER.TIME$(09)="090" ' 1 HOURS/30 MINUTES
0ED9 2324 PROVIDER.TIME$(10)="120" ' 2 HOURS

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

0EE2 2324 PROVIDER.TIMES(11)="150" ' 2 HOURS/30 MINUTES
0EEB 2324 PROVIDER.TIMES(12)="180" ' 3 HOURS
0EF4 2324 PROVIDER.TIMES(13)="210" ' 3 HOURS/30 MINUTES
0EFD 2324 PROVIDER.TIMES(14)="240" ' 4 HOURS
0F06 2324 PROVIDER.TIMES(15)="270" ' 4 HOURS/30 MINUTES
0F0F 2324 PROVIDER.TIMES(16)="300" ' 5 HOURS
0F18 2324 PROVIDER.TIMES(17)="330" ' 5 HOURS/30 MINUTES
0F21 2324 PROVIDER.TIMES(18)="360" ' 6 HOURS
0F2A 2324 PROVIDER.TIMES(19)="390" ' 6 HOURS/30 MINUTES
0F33 2324 PROVIDER.TIMES(20)="420" ' 7 HOURS
0F3C 2324 PROVIDER.TIMES(21)="450" ' 7 HOURS/30 MINUTES
0F45 2324 PROVIDER.TIMES(22)="480" ' 8 HOURS
0F4E 2324
0F4E 2324 REM YES/NO TABLE
0F4E 2324 YNS(0)=" " : YNS(1)="Y" : YNS(2)="N" : YNS(3)="X"
0F72 2324
0F72 2324 REM *** TABLE OF OTHER CODES ***
0F72 2324 SOCWOS(1)="02000" : RHEUMS(01)="06002" : OPTOMS(01)="03001"
0F8D 2324 RHEUMS(02)="06003" : OPTOMS(02)="03002"
0F9F 2324 RHEUMS(03)="06004" : OPTOMS(03)="03003"
0F81 2324 PSCOLS(01)="04032" : RHEUMS(04)="06005" : OPTOMS(04)="03004"
0FCC 2324 PSCOLS(02)="04033" : RHEUMS(05)="06011" : OPTOMS(05)="03005"
0FE7 2324 PSCOLS(03)="04034" : RHEUMS(06)="06010" : OPTOMS(06)="03006"
1002 2324 PSCOLS(04)="04026" : RHEUMS(07)="06007" : OPTOMS(07)="03007"
101D 2324 PSCOLS(05)="04031" : RHEUMS(08)="06001" : OPTOMS(08)="03008"
1038 2324 PSCOLS(06)="04028" : RHEUMS(09)="06006" : OPTOMS(09)="03009"
1053 2324 PSCOLS(07)="04029" : RHEUMS(10)="06008" : OPTOMS(10)="03010"
106E 2324 PSCOLS(08)="04023" : RHEUMS(11)="06009" : OPTOMS(11)="03011"
1089 2324 PSCOLS(09)="04025" : OPTOMS(12)="03012"
1098 2324 PSCOLS(10)="04024" : OPTOMS(13)="03013"
10AD 2324 PSCOLS(11)="04027" : PSYCHS(01)="04026" : OPTOMS(14)="03014"
10C8 2324 PSCOLS(12)="04020" : PSYCHS(02)="04028" : OPTOMS(15)="03015"
10E3 2324 PSCOLS(13)="04030" : PSYCHS(03)="04029" : OPTOMS(16)="03016"
10FE 2324 PSCOLS(14)="04021" : PSYCHS(04)="04023" : OPTOMS(17)="03017"
1119 2324 PSCOLS(15)="04022" : PSYCHS(05)="04025" : OPTOMS(18)="03018"
1134 2324 PSCOLS(16)="04040" : PSYCHS(06)="04024" : OPTOMS(19)="03019"
114F 2324 PSCOLS(17)="04041" : PSYCHS(07)="04027" : OPTOMS(20)="03020"
116A 2324 PSCOLS(18)="04000" : PSYCHS(08)="04020" : OPTOMS(21)="03021"
1185 2324 PSCOLS(19)="04001" : PSYCHS(09)="04030"
1197 2324 PSCOLS(20)="04002" : PSYCHS(10)="04021"
11A9 2324 PSCOLS(21)="04003" : PSYCHS(11)="04022" : DIALYS(01)="06101"
11C4 2324 PSCOLS(22)="04004" : PSYCHS(12)="04040" : DIALYS(02)="06102"
11DF 2324 PSCOLS(23)="04005" : PSYCHS(13)="04041" : DIALYS(03)="06103"
11FA 2324 PSCOLS(24)="04006" : PSYCHS(14)="04000" : DIALYS(04)="06104"
1215 2324 PSCOLS(25)="04007" : PSYCHS(15)="04001" : DIALYS(05)="06105"
1230 2324 PSCOLS(26)="04008" : PSYCHS(16)="04002" : DIALYS(06)="06106"
1248 2324 PSCOLS(27)="04009" : PSYCHS(17)="04003" : DIALYS(07)="06107"
1266 2324 PSCOLS(28)="04010" : PSYCHS(18)="04004" : DIALYS(08)="06108"
1281 2324 PSCOLS(29)="04011" : PSYCHS(19)="04005" : DIALYS(09)="06109"
129C 2324 PSCOLS(30)="04012" : PSYCHS(20)="04006" : DIALYS(10)="06110"
12B7 2324 PSCOLS(31)="04013" : PSYCHS(21)="04007" : DIALYS(11)="06111"
12D2 2324 PSCOLS(32)="04014" : PSYCHS(22)="04008" : DIALYS(12)="06112"
12ED 2324 PSCOLS(33)="04015" : PSYCHS(23)="04009" : DIALYS(13)="06113"
1308 2324 PSCOLS(34)="04016" : PSYCHS(24)="04010" : DIALYS(14)="06114"

```

Offset	Data	Source Line
1323	2324	PSYCH\$(25)="04011" : DIALYS(15)="06115"
1335	2324	PSYCH\$(26)="04012" : DIALYS(16)="06116"
1347	2324	PSYCH\$(27)="04013" : DIALYS(17)="06117"
1359	2324	PSYCH\$(28)="04014" : DIALYS(18)="06118"
1368	2324	PSYCH\$(29)="04015" : DIALYS(19)="06119"
1370	2324	PSYCH\$(30)="04016" : DIALYS(20)="06120"
138F	2324	DIALYS(21)="06121"
1398	2324	PHYTH\$(01)="05001" : DIALYS(22)="06122"
13AA	2324	PHYTH\$(02)="05002" : DIALYS(23)="06123"
13BC	2324	PHYTH\$(03)="05003" : PSYAS\$(1)="02567" : DIALYS(24)="06124"
13D7	2324	PHYTH\$(04)="05004" : PSYAS\$(2)="02565"
13E9	2324	PHYTH\$(05)="05005" : PSYAS\$(3)="02566" : NEUROS(14)="08013"
1404	2324	PHYTH\$(06)="05006" : PSYAS\$(4)="02568" : NEUROS(15)="08014"
141F	2324	PHYTH\$(07)="05007" : NEUROS(16)="08015"
1431	2324	PHYTH\$(08)="05008" : NEUROS(01)="08000" : NEUROS(17)="08016"
144C	2324	PHYTH\$(09)="05009" : NEUROS(02)="08001" : NEUROS(18)="08017"
1467	2324	PHYTH\$(10)="05010" : NEUROS(03)="08002" : NEUROS(19)="08018"
1482	2324	PHYTH\$(11)="05011" : NEUROS(04)="08003" : NEUROS(20)="08019"
1490	2324	PHYTH\$(12)="05012" : NEUROS(05)="08004" : NEUROS(21)="08020"
1488	2324	PHYTH\$(13)="05013" : NEUROS(06)="08005" : NEUROS(22)="08021"
1403	2324	PHYTH\$(14)="05014" : NEUROS(07)="08006" : NEUROS(23)="08022"
14EE	2324	PHYTH\$(15)="05015" : NEUROS(08)="08007" : NEUROS(24)="08023"
1509	2324	PHYTH\$(16)="05016" : NEUROS(09)="08008" : NEUROS(25)="08024"
1524	2324	PHYTH\$(17)="05017" : NEUROS(10)="08009" : NEUROS(26)="08025"
153F	2324	PHYTH\$(18)="05018" : NEUROS(11)="08010" : NEUROS(27)="08026"
155A	2324	PHYTH\$(19)="05019" : NEUROS(12)="08011" : NEUROS(28)="08027"
1575	2324	NEUROS(13)="08012" : NEUROS(29)="08028"
1587	2324	
1587	2324	ORTHOS(00)=" " : ORTHOS(37)="05537" : ORTHOS(74)="05574"
15A2	2324	ORTHOS(01)="05501" : ORTHOS(38)="05538" : ORTHOS(75)="05575"
15B0	2324	ORTHOS(02)="05502" : ORTHOS(39)="05539" : ORTHOS(76)="05576"
1508	2324	ORTHOS(03)="05503" : ORTHOS(40)="05540" : ORTHOS(77)="05577"
15F3	2324	ORTHOS(04)="05504" : ORTHOS(41)="05541" : ORTHOS(78)="05578"
160E	2324	ORTHOS(05)="05505" : ORTHOS(42)="05542" : ORTHOS(79)="05579"
1629	2324	ORTHOS(06)="05506" : ORTHOS(43)="05543" : ORTHOS(80)="05580"
1644	2324	ORTHOS(07)="05507" : ORTHOS(44)="05544" : ORTHOS(81)="05581"
165F	2324	ORTHOS(08)="05508" : ORTHOS(45)="05545" : ORTHOS(82)="05582"
167A	2324	ORTHOS(09)="05509" : ORTHOS(46)="05546" : ORTHOS(83)="05583"
1695	2324	ORTHOS(10)="05510" : ORTHOS(47)="05547" : ORTHOS(84)="05584"
1680	2324	ORTHOS(11)="05511" : ORTHOS(48)="05548" : ORTHOS(85)="05585"
16CB	2324	ORTHOS(12)="05512" : ORTHOS(49)="05549" : ORTHOS(86)="05586"
16E6	2324	ORTHOS(13)="05513" : ORTHOS(50)="05550" : ORTHOS(87)="05587"
1701	2324	ORTHOS(14)="05514" : ORTHOS(51)="05551" : ORTHOS(88)="05588"
171C	2324	ORTHOS(15)="05515" : ORTHOS(52)="05552" : ORTHOS(89)="05589"
1737	2324	ORTHOS(16)="05516" : ORTHOS(53)="05553" : ORTHOS(90)="05590"
1752	2324	ORTHOS(17)="05517" : ORTHOS(54)="05554" : ORTHOS(91)="05591"
1760	2324	ORTHOS(18)="05518" : ORTHOS(55)="05555" : ORTHOS(92)="05592"
1788	2324	ORTHOS(19)="05519" : ORTHOS(56)="05556" : ORTHOS(93)="05593"
17A3	2324	ORTHOS(20)="05520" : ORTHOS(57)="05557" : ORTHOS(94)="05594"
17BE	2324	ORTHOS(21)="05521" : ORTHOS(58)="05558" : ORTHOS(95)="05595"
17D9	2324	ORTHOS(22)="05522" : ORTHOS(59)="05559" : ORTHOS(96)="05596"
17F4	2324	ORTHOS(23)="05523" : ORTHOS(60)="05560" : ORTHOS(97)="05597"
180F	2324	ORTHOS(24)="05524" : ORTHOS(61)="05561" : ORTHOS(98)="05598"
182A	2324	ORTHOS(25)="05525" : ORTHOS(62)="05562" : ORTHOS(99)="05599"

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

1845 2324 ORTHOS(26)="05526" : ORTHOS(63)="05563" : ORTHOS(100)="05600"
1860 2324 ORTHOS(27)="05527" : ORTHOS(64)="05564" : ORTHOS(101)="05601"
1878 2324 ORTHOS(28)="05528" : ORTHOS(65)="05565" : ORTHOS(102)="05602"
1896 2324 ORTHOS(29)="05529" : ORTHOS(66)="05566" : ORTHOS(103)="05603"
18B1 2324 ORTHOS(30)="05530" : ORTHOS(67)="05567" : ORTHOS(104)="05604"
18CC 2324 ORTHOS(31)="05531" : ORTHOS(68)="05568" : ORTHOS(105)="05605"
18E7 2324 ORTHOS(32)="05532" : ORTHOS(69)="05569" : ORTHOS(106)="05606"
1902 2324 ORTHOS(33)="05533" : ORTHOS(70)="05570" : ORTHOS(107)="05607"
1910 2324 ORTHOS(34)="05534" : ORTHOS(71)="05571"
192F 2324 ORTHOS(35)="05535" : ORTHOS(72)="05572"
1941 2324 ORTHOS(36)="05536" : ORTHOS(73)="05573"
1953 2324
1953 2324 REM *****
1953 2324 PNUM=VAL(PGMID$)/10
1965 2326 REM LENGTH OF STRING RECEIVED FROM THE OMR....
1965 2326 HEADER = 21
196C 2328 RESPONSE= NRESP(PNUM)
197A 232A RECORDLENGTH = HEADER + RESPONSE
1985 232C
1985 232C N.PROC = PROCN(PNUM) ' NUMBER OF PROCEDURES FOR THIS FORM
1993 232E N.DIAG.COL=4 ' NUMBER OF DX COLUMNS ON THIS FORM
199A 2330 IF CC4(PNUM)=0 THEN N.DIAG.COL=N.DIAG.COL-1
1983 2330 IF CC3(PNUM)=0 THEN N.DIAG.COL=N.DIAG.COL-1
19CC 2330 IF CC2(PNUM)=0 THEN N.DIAG.COL=N.DIAG.COL-1
19E5 2330
19E5 2330 DATFIL$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
19EE 2334 BTIMES=TIMES 'SCAN START TIME
19F7 2338
19F7 2338 REM *** ENCOUNTER FORM PROCEDURE TABLE ***
19F7 2338 F.NAMES="RACPROC." + PGMID$
1A05 233C
1A05 233C OPEN F.NAMES$ FOR INPUT AS #3
1A16 233C FOR I600=0 TO 125
1A1C 233C INPUT #3,PROCD$(I600)
1A34 233E IF PROCD$(I600)="ZZZZZ" THEN GOTO 4
1A50 233E NEXT I600
1A5F 233E 4 CLOSE #3
1A66 233E
1A66 233E REM *** ENCOUNTER FORM DIAGNOSIS TABLE ***
1A66 233E F.NAMES="RACDIAG." + PGMID$
1A74 233E
1A74 233E OPEN F.NAMES$ FOR INPUT AS #3
1A85 233E FOR I600 = 0 TO 225
1A8B 233E INPUT #3,DIAGN.TAB$(I600)
1AA3 233E IF DIAGN.TAB$(I600)="ZZZZZ" THEN GOTO 6
1ABF 233E NEXT I600
1ACF 233E 6 CLOSE #3
1AD6 233E
1AD6 233E REM INCLUDE: 'UCACAMP.OPT' INCLUDE THE OUTP UCA VALIDATE TABLE
1AD6 233E REM INCLUDE: 'UCACAMP.IPT' INCLUDE THE INP UCA VALIDATE TABLE
1AD6 233E
1AD6 233E REM $PAGE

```

Offset	Data	Source Line
1AD6	233E	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
1A0B	233E	CLS
1ADF	233E	GOSUB 7000 'PRINT SCREEN HEADING
1AE4	233E	
1AE4	233E	REM *****
1AE4	233E	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
1AE4	233E	REM *****
1AE4	233E	REM
1AE4	233E	OPEN DATFIL\$ FOR APPEND AS #1
1AF6	233E	
1AF6	233E	REM *****
1AF6	233E	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
1AF6	233E	REM *****
1AF6	233E	LPRINT CHR\$(15);
1B01	233E	WIDTH "LPT1:",160
1B0B	233E	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
1B17	2340	COLOR 14
1B1E	2340	LOCATE 11,26 : PRINT "BASE ENCOUNTER FORM "
1B33	2340	COLOR FORE,BACK,BORD
1B49	2340	
1B49	2340	REM *****
1B49	2340	REM **** COMMUNICATIONS SETUP ****
1B49	2340	REM *****
1B49	2340	REM PROTOCOL
1B49	2340	GOSUB 9001
1B4E	2340	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
1B68	2344	
1B68	2344	REM START SCANNER (S1)
1B68	2344	CNTRLOPT = 1 : GOSUB 9010
1B74	2346	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
1B8E	2346	
1B8E	2346	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
1BA3	2346	READTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
1BAA	2348	
1BAA	2348	REM *****
1BAA	2348	REM *** SET SCAN SHEET CALL ***
1BAA	2348	REM *****
1BAA	2348	REM
1BAA	2348	
1BAA	2348	10 REM - RETURN POINT TO READ NEXT SHEET
1BAB	2348	
1BAB	2348	A\$=INKEY\$
1BB4	234C	IF A\$=CHR\$(27) THEN GOTO 25000
1BCA	234C	
1BCA	234C	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
1BCF	234C	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
1BEB	2350	
1BEB	2350	TEXT\$="" 'CLEAR THE INPUT AREA
1BF4	2350	GOSUB 8000 'DECODE HEADER
1BF9	2350	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
1BFE	2350	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
1C03	2350	LITHO\$ = MID\$(TEXT\$,22,8)
1C15	2354	GOSUB 8070 'CHECK FOR SCANNER ERRORS
1C1A	2354	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

Offset	Data	Source Line
1C1F	2354	
1C1F	2354	REM \$INCLUDE: 'RACREAD.MOD' INCLUDE THE READ FORM REFORMAT/EDIT MOD
1C1F	2354	REM *****
1C1F	2354	REM **** AMBULATORY CARE INFORMATION SYSTEM 10 MAR 87 ****
1C1F	2354	REM **** D R BOLLING ****
1C1F	2354	REM **** MODULE NAME : RACREAD.MOD ****
1C1F	2354	REM **** SCANNER PROGRAM # : GENERAL FORM ****
1C1F	2354	REM ****
1C1F	2354	REM **** PURPOSE : REFORMAT/EDIT THE FORM ****
1C1F	2354	REM ****
1C1F	2354	REM *****
1C1F	2354	REM **** RESERVED LINE NUMBERS 100-199 ****
1C1F	2354	REM *****
1C1F	2354	
1C1F	2354	N.ERR =0 'COUNTS THE NUMBER OF ERRORS
1C26	2356	
1C26	2356	REM *** LITHO CODE DONE IN BAS PROGRAM ***
1C26	2356	REM *** CLINIC ID (PREFIX + CODE) ***
1C26	2356	100 CL1.COD\$=CLIDEF\$(PNUM) 'DEFAULT CLINIC CODE
1C38	235A	
1C38	235A	REM ... CLINIC CODE WHEN SINGLE BUBBLED ...
1C38	235A	IF MID\$(TEXT\$,34,1)=" " THEN 102
1C50	235A	X=VAL(MID\$(TEXT\$,34,1))
1C66	235C	'
1C66	235C	' SET PTH=1 IF PHYS THER FORM NEEDS MULTIPLE RECORD OUTPUT
1C66	235C	IF LEFT\$(PGMID\$,2)="68" AND X=2 THEN PTH=1 ELSE PTH=0
1CA0	235E	'
1CA0	235E	X=(X-1)*4+1 'CALCULATE OFFSET
1CB0	235E	CL1.COD\$=MID\$(CLIBUB\$(PNUM),X,4)
1CCC	235E	GOTO 104
1CD0	235E	
1CD0	235E	REM ... CLINIC CODE WHEN HAND CODED ...
1CD0	235E	102 IF MID\$(TEXT\$,30,4)=" " THEN 104
1CE8	235E	CK.X=VAL(MID\$(TEXT\$,30,1))
1CFE	2360	CK.COD\$=MID\$(TEXT\$,31,3)
1D10	2364	REM GOSUB 5500 'OUTP UCA CODE CHECK
1D10	2364	REM IF RT.5500 = 0 THEN GOTO 102
1D10	2364	REM N.ERR = N.ERR + 1
1D10	2364	REM ED.MSG\$(N.ERR)="INVALID CLINIC CODE"
1D10	2364	
1D10	2364	CL1.COD\$=CLINIC1.PFX\$(CK.X) + CK.COD\$ 'HAND CODED
1D28	2364	
1D28	2364	REM VISIT DATE
1D28	2364	104 CM\$=MID\$(DATES,1,2) 'CURRENT MONTH
1D3A	2368	YR\$=MID\$(DATES,9,2) 'CURRENT YEAR
1D4C	236C	XS=MID\$(TEXT\$,35,4) 'MONTH AND DAY
1D5E	2370	IF LEFT\$(XS,2)<=CM\$ THEN 105 'OK, USE THIS YEAR
1D73	2370	YR\$=RIGHT\$(STR\$(VAL(YR\$)-1),2) 'USE LAST YEAR
1D93	2370	105 VIDATES=YR\$+XS
1DA1	2374	D1\$=MID\$(XS,3,1)
1DB3	2378	D2\$=MID\$(XS,4,1)
1DC5	237C	IF D1\$=" " AND D2\$<>" " THEN RT.5000=2 : GOTO 106
1DF3	237E	IF D1\$<>" " AND D2\$=" " THEN RT.5000=2 : GOTO 106
1E21	237E	'EDIT VISIT DATE

Offset	Data	Source Line
1E21	237E	CK.5000\$=VIDATES
1E2A	2382	GOSUB 5000 'DATE CHECK
1E2F	2382	IF RT.5000=0 THEN GOTO 107
1E3E	2382	106 N.ERR=N.ERR+1
1E46	2382	ED.MSG\$(N.ERR)="VISIT DATE" + DATEERR\$(RT.5000)
1E6C	2382	
1E6C	2382	REM *** PRIMARY PROVIDER ***
1E6C	2382	107 PPROV.PFX\$=MID\$(TEXT\$,39,1)
1E7E	2386	PPROV.NUM\$=MID\$(TEXT\$,40,4)
1E90	238A	
1E90	238A	REM *** PATIENT SSN ***
1E90	238A	108 SSN\$ = MID\$(TEXT\$,44,9)
1EA2	238E	
1EA2	238E	REM *** FAMILY MEMBER PREF ***
1EA2	238E	110 FMEMPS=MID\$(TEXT\$,53,2)
1EB4	2392	
1EB4	2392	REM *** FORM NUMBER ***
1EB4	2392	FRMNS=LEFT\$(PGMID\$,2)
1EC3	2396	
1EC3	2396	REM *** VISIT COUNT ***
1EC3	2396	112 VCNT\$="1"
1ECC	239A	TEMPS=MID\$(TEXT\$,55,1)
1EDE	239E	IF TEMPS="1" THEN VCNT\$="0"
1EF5	239E	IF TEMPS>"1" THEN VCNT\$=TEMPS
1F0C	239E	
1F0C	239E	REM *** PRIMARY PROVIDER TIME ***
1F0C	239E	113 X=VAL(MID\$(TEXT\$,56,2))
1F22	239E	PPROV.TIMS=PROVIDER.TIMES(X)
1F34	23A2	
1F34	23A2	REM *** SECONDARY PROVIDER ***
1F34	23A2	SPROV.PFX\$=MID\$(TEXT\$,58,1)
1F46	23A6	SPROV.NUM\$=MID\$(TEXT\$,59,4)
1F58	23AA	IF SPROV.PFX\$+SPROV.NUM\$ = " " THEN SPBL=1 ELSE SPBL=0
1F7D	23AC	IF SPROV.PFX\$=" " AND SPROV.NUM\$<>" " THEN 114
1FA3	23AC	IF SPROV.PFX\$<>" " AND SPROV.NUM\$=" " THEN 114
1FC9	23AC	GOTO 115
1FCD	23AC	
1FCD	23AC	114 N.ERR = N.ERR + 1
1FD5	23AC	ED.MSG\$(N.ERR)="PROV 2 CODE MISSING PREFIX OR NUMBER"
1FE9	23AC	
1FE9	23AC	
1FE9	23AC	REM *** SECONDARY PROVIDER TIME ***
1FE9	23AC	115 X=VAL(MID\$(TEXT\$,63,2))
1FFF	23AC	SPROV.TIMS=PROVIDER.TIMES(X)
2011	23B0	REM IS THERE A TIME AND NO SEC PROV CODED?
2011	23B0	IF SPROV.TIMS <> "000" AND SPBL = 1 THEN 116 'GO IF YES
2034	23B0	REM IS THERE NO TIME AND A SEC PROV CODED?
2034	23B0	IF SPROV.TIMS = "000" AND SPBL = 0 THEN 117 'GO IF YES
2057	23B0	GOTO 118
2058	23B0	
2058	23B0	116 N.ERR = N.ERR + 1
2063	23B0	ED.MSG\$(N.ERR)="TIME CODED WITH NO SEC PROV CODED"
2077	23B0	GOTO 118
2078	23B0	

Offset	Data	Source Line
207B	23B0	117 N.ERR = N.ERR + 1
2083	23B0	ED.MSG\$(N.ERR)="NO PROV 2 TIME"
2097	23B0	
2097	23B0	REM *** REASON FOR SECONDARY PROVIDER ***
2097	23B0	118 SPROV.REAS=MID\$(TEXT\$,65,1)
20A9	23B4	REM IS THERE A REASON AND NO SEC PROV CODED?
20A9	23B4	IF SPROV.REAS <> " " AND SPBL = 1 THEN 119 'GO IF YES
20CC	23B4	REM IS THERE NO REASON AND A SEC PROV CODED?
20CC	23B4	IF SPROV.REAS = " " AND SPBL = 0 THEN 120 'GO IF YES
20EF	23B4	GOTO 122
20F3	23B4	
20F3	23B4	119 N.ERR = N.ERR + 1
20FB	23B4	ED.MSG\$(N.ERR)="REASON CODED WITH NO SEC PROV CODED"
210F	23B4	GOTO 122
2113	23B4	
2113	23B4	120 N.ERR = N.ERR + 1
211B	23B4	ED.MSG\$(N.ERR)="NO PROV 2 REASON"
212F	23B4	
212F	23B4	REM *** IF NOT SCHEDULED ***
212F	23B4	122 X=VAL(MID\$(TEXT\$,66,1))
2145	23B4	NOTSCS=RIGHT\$(STR\$(X+1),1) 'DEFAULT IS 1
2159	23B8	
2159	23B8	REM *** IF NOT CLINIC ***
2159	23B8	124 X=VAL(MID\$(TEXT\$,67,1))
216F	23B8	NOTCLS=RIGHT\$(STR\$(X+1),1) 'DEFAULT IS 1
2183	23BC	
2183	23BC	REM *** REFERRAL CODE ***
2183	23BC	126 CX=X=VAL(MID\$(TEXT\$,68,1))
2199	23BC	CK.CODS=MID\$(TEXT\$,69,3)
21AB	23BC	REM IF CK.X=0 AND CK.CODS=" " THEN GOTO 142 'MAY NOT BE CODED
21AB	23BC	REM GOSUB 5600 'VALIDATE INP UCA CODE
21AB	23BC	REM IF RT.5600 = 0 THEN GOTO 142
21AB	23BC	REM N.ERR = N.ERR + 1
21AB	23BC	REM ED.MSG\$(N.ERR)="INVALID REFERRAL CODE"
21AB	23BC	
21AB	23BC	128 REF.PFX\$=CLINIC2.PFX\$(CK.X)
21BD	23C0	REF.COD\$=CK.COD\$
21C6	23C4	
21C6	23C4	REM ... OTHER REFERRAL CODES (FORM 680 -PHYS THER ONLY)
21C6	23C4	IF FRMNS<>"68" THEN 129 'DO ONLY FOR 680
21D4	23C4	X=VAL(MID\$(TEXT\$,76,2))
21EA	23C4	IF X=0 THEN 129 'NO CODE BUBBLED
21F5	23C4	REF.PFX\$=MID\$(REFCOS(X),1,1)
220F	23C4	REF.COD\$=MID\$(REFCOS(X),2,3)
222A	23C4	
222A	23C4	129 IF FRMNS<>"65" THEN 130 'DO ONLY FOR 650
2238	23C4	X=VAL(MID\$(TEXT\$,76,2))
224E	23C4	IF X=0 THEN 130 'NO CODE BUBBLED
2259	23C4	REF.PFX\$=MID\$(REFCOS(X),1,1)
2273	23C4	REF.COD\$=MID\$(REFCOS(X),2,3)
228E	23C4	
228E	23C4	REM *** JOB RELATED VISIT ***
228E	23C4	130 RELAT.VISS="N"
2297	23C8	IF MID\$(TEXT\$,72,1)="1" THEN RELAT.VISS="Y"

Offset	Data	Source Line
2288	23C8	
2288	23C8	REM *** MIL ONLY DUTY ***
2288	23C8	132 MILDUTS=MIDS(TEXT\$,73,1)
22CA	23CC	
22CA	23CC	REM *** MIL ONLY QTRS ***
22CA	23CC	134 MILQTRS=MIDS(TEXT\$,74,1)
22DC	2300	
22DC	2300	REM *** MIL ONLY PROF ***
22DC	2300	136 MILPROS=MIDS(TEXT\$,75,1)
22EE	2304	
22EE	2304	REM *** NOT AVAILABLE ***
22EE	2304	138 NAVAILS=MIDS(TEXT\$,76,1)
2300	2308	
2300	2308	REM *** SPEC PREASSIGNED CLINIC ***
2300	2308	140 INP.STOS=MIDS(TEXT\$,81,12)
2312	230C	GOSUB 5700 'CONVERT ARRAY
2317	230C	SPE.BUFS=BUF.STOS 'UP TO 12 TWO DIGIT CODES
2320	23E4	
2320	23E4	REM *** PROVIDER 2 ADDL PROC 1 ***
2320	23E4	144 PR2PRC1\$="N"
2329	23E8	IF MIDS(TEXT\$,93,1)="1" THEN PR2PRC1\$="Y"
234A	23E8	IF PR2PRC1\$="Y" AND SPBL=1 THEN 145
2360	23E8	GOTO 146
2371	23E8	
2371	23E8	145 N.ERR = N.ERR + 1
2379	23E8	ED.MSG\$(N.ERR)="ADDL PROC CODE FOR SEC PROV BUT NO SEC PROV CODED"
2380	23E8	
2380	23E8	REM *** ADDITIONAL PROCEDURE 1 ***
2380	23E8	146 ADDP1\$=MIDS(TEXT\$,94,5)
239F	23EC	
239F	23EC	REM *** PROVIDER 2 ADDL PROC 2 ***
239F	23EC	148 PR2PRC2\$="N"
23A8	23F0	IF MIDS(TEXT\$,99,1)="1" THEN PR2PRC2\$="Y"
23C9	23F0	IF PR2PRC2\$="Y" AND SPBL=1 THEN 149
23EC	23F0	GOTO 150
23F0	23F0	
23F0	23F0	149 N.ERR = N.ERR + 1
23F8	23F0	ED.MSG\$(N.ERR)="ADDL PROC CODE FOR SEC PROV BUT NO SEC PROV CODED"
240C	23F0	
240C	23F0	REM *** ADDITIONAL PROCEDURE 2 ****
240C	23F0	150 ADDP2\$=MIDS(TEXT\$,100,5)
241E	23F4	
241E	23F4	REM *** ADMITTED ***
241E	23F4	ADMIT\$="N"
2427	23F8	IF MIDS(TEXT\$,105,1)="1" THEN ADMIT\$="Y"
2448	23F8	
2448	23F8	REM *** EDIT FOR MIL ONLY BOX ***
2448	23F8	IF ADMIT\$="N" THEN 155 'OK, NO NEED TO CHECK
2456	23F8	IF MILDUTS=" " AND MILQTRS=" " THEN 155 'OK
2479	23F8	
2479	23F8	152 REM ... ERROR DUTY, QTRS AND ADMITTED CANNOT ALL BE CODED
247A	23F8	N.ERR = N.ERR + 1
2492	23F8	ED.MSG\$(N.ERR)="DUTY OR QTRS CANNOT BE CODED WITH ADMITTED"
2496	23F8	

Offset	Data	Source Line
2496	23F8	REM *** UNLISTED PRIMARY DX ***
2496	23F8	155 X=VAL(MID\$(TEXT\$,106,1))
24AC	23F8	IF X=0 THEN PRIMDX\$=""
24C0	23FC	IF X=1 THEN PRIMDX\$="V"
24D4	23FC	IF X=2 THEN PRIMDX\$="S"
24E8	23FC	DXTMP\$=MID\$(TEXT\$,107,5)
24FA	2400	IF DXTMP\$="" THEN GOTO 158
250C	2400	
250C	2400	REM REMOVE LEADING BLANKS
250C	2400	156 IF LEFT\$(DXTMP\$,1)="" THEN DXTMP\$=RIGHT\$(DXTMP\$,4)+" ":GOTO 156
253B	2400	
253B	2400	158 LASTC\$=RIGHT\$(DXTMP\$,1) 'GET LAST CHAR
254A	2404	IF LASTC\$<>" " AND LASTC\$<>"0" THEN PRIMDX\$="" 'REMOVE V OR S
2576	2404	PRIMDX\$=LEFT\$(PRIMDX\$+DXTMP\$,5) 'GET 5 CHAR ONLY
2588	2404	
2588	2404	REM *** UNLISTED SECONDARY DX ***
2588	2404	160 X=VAL(MID\$(TEXT\$,112,1))
259E	2404	IF X=0 THEN SECDX\$=""
25B2	2408	IF X=1 THEN SECDX\$="V"
25C6	2408	IF X=2 THEN SECDX\$="S"
25DA	2408	DXTMP\$=MID\$(TEXT\$,113,5)
25EC	2408	IF DXTMP\$="" THEN GOTO 164
25FE	2408	
25FE	2408	REM REMOVE LEADING BLANKS
25FE	2408	162 IF LEFT\$(DXTMP\$,1)="" THEN DXTMP\$=RIGHT\$(DXTMP\$,4)+" ":GOTO 162
262D	2408	
262D	2408	164 LASTC\$=RIGHT\$(DXTMP\$,1) 'GET LAST CHAR
263C	2408	IF LASTC\$<>" " AND LASTC\$<>"0" THEN SECDX\$="" 'REMOVE V OR S
2668	2408	SECDX\$=LEFT\$(SECDX\$+DXTMP\$,5) 'GET 5 CHAR ONLY
267A	2408	
267A	2408	REM *** FOLLOW UP/RULE OUT ***
267A	2408	166 FU.ROS=MID\$(TEXT\$,118,1)
268C	240C	
268C	240C	REM *** EVALUATION/SERV/PROC PROV 1 ***
268C	240C	168 INP.STO\$=MID\$(TEXT\$,119,N.PROC)
269F	240C	ESP.BUF1\$=""
26A8	2410	N.STO=1
26AF	2412	
26AF	2412	170 X.STO=INSTR(N.STO,INP.STO\$,"1")
26C1	2414	IF X.STO=0 THEN GOTO 172 'THATS ALL
26D0	2414	N.STO=X.STO + 1 'NEXT STARTING POINT
26D8	2414	ESP.BUF1\$=ESP.BUF1\$ + PROCED\$(X.STO) 'ADD CODE TO BUFFER BY FIVES
26ED	2414	IF N.STO <= LEN(INP.STO\$) THEN GOTO 170
2700	2414	
2700	2414	REM *** EVALUATION/SERV/PROC PROV 2 ***
2700	2414	172 INP.STO\$=MID\$(TEXT\$,119+N.PROC,N.PROC)
2717	2414	ESP.BUF2\$=""
2720	2418	N.STO=1
2727	2418	
2727	2418	174 X.STO=INSTR(N.STO,INP.STO\$,"1")
2739	2418	IF X.STO=0 THEN GOTO 176 'THATS ALL
2748	2418	IF SPROV.NUM\$ = " " THEN 175 'NEED SEC PROV CODED
2756	2418	N.STO=X.STO + 1 'NEXT STARTING POINT
275E	2418	ESP.BUF2\$=ESP.BUF2\$ + PROCED\$(X.STO) 'ADD CODE TO BUFFER BY FIVES

Offset	Data	Source Line
2773	2418	IF N.STO <= LEN(INP.STO\$) THEN GOTO 174
2786	2418	GOTO 176
278A	2418	
278A	2418	175 N.ERR = N.ERR + 1
2792	2418	ED.MSG\$(N.ERR)="PROC CODE FOR SEC PROV BUT NO SEC PROV CODED"
27A6	2418	
27A6	2418	176 T.POS = 119 + 2*N.PROC
27B4	241A	
27B4	241A	REM *** OTHER CODES ***
27B4	241A	IF NOTH(PNUM)=0 THEN 184 'SKIP THIS IF NO CODES
27C5	241A	178 INP.STO\$=MID\$(TEXT\$,T.POS,NOTH(PNUM))
27DF	241A	OTH.BUF\$=""
27E8	241E	N.STO=1
27EF	241E	
27EF	241E	180 X.STO=INSTR(N.STO,INP.STO\$,"1")
2801	241E	IF X.STO=0 THEN GOTO 182 'THATS ALL
2810	241E	N.STO=X.STO + 1 'NEXT STARTING POINT
2818	241E	IF PNUM =54 THEN OX\$=DIALYS\$(X.STO) 'NEPHROLOGY/DIALYSIS CODES
2835	2422	IF PNUM =56 THEN OX\$=NEUROS\$(X.STO) 'NEUROSURGERY CODES
2852	2422	IF PNUM =62 THEN OX\$=OPTOMS\$(X.STO) 'OPTOMETRY CODES
286F	2422	IF PNUM =63 THEN OX\$=ORTHO\$(X.STO) 'ORTHO APPL CODES
288C	2422	IF PNUM =68 THEN OX\$=PHYTH\$(X.STO) 'PHYSICAL THERAPY CODES
28A9	2422	IF PNUM =73 THEN OX\$=PSYCH\$(X.STO) 'PSYCHIATRY CODES
28C6	2422	IF PNUM =74 THEN OX\$=PSCOLS\$(X.STO) 'PSYCHOLOGY CODES
28E3	2422	IF PNUM =77 THEN OX\$=RHEUM\$(X.STO) 'RHEUMATOLOGY CODES
2900	2422	IF PNUM =78 THEN OX\$=SOCWOS\$(X.STO) 'SOCIAL WORK CODES
2910	2422	OTH.BUF\$=OTH.BUF\$ + OX\$ 'ADD CODE TO BUFFER BY FIVES
2929	2422	IF N.STO <= LEN(INP.STO\$) THEN GOTO 180
293C	2422	
293C	2422	182 T.POS = T.POS + NOTH(PNUM)
294E	2422	184 REM
294F	2422	
294F	2422	REM *** PRIMARY DX ***
294F	2422	REM * IF OTHER PRIM DX IS CODED THEN SKIP THIS SECTION *
294F	2422	186 IF PRIMOX\$<>" " THEN GOTO 192
2961	2422	X.POS=T.POS 'STARTING POSITION
2968	2424	C1SIZ=CC1(PNUM): C3SIZ=CC3(PNUM) 'NO. OF ITEMS IN EACH COL
2984	2428	C2SIZ=CC2(PNUM): C4SIZ=CC4(PNUM)
29A0	242C	GOSUB 5800 'GET POSITION
29A5	242C	IF X.FIN=0 THEN GOTO 188
29B4	242E	IF RT.5800=0 THEN GOTO 190
29C3	2430	N.ERR=N.ERR + 1
29CB	2430	ED.MSG\$(N.ERR)="PRIMARY DX HAS MULTIPLE CODES"
29DF	2430	GOTO 190
29E3	2430	
29E3	2430	188 N.ERR=N.ERR + 1
29EB	2430	ED.MSG\$(N.ERR)="PRIMARY DX NOT CODED"
29FF	2430	
29FF	2430	190 PRIMDX\$=DIAGN.TAB\$(X.FIN)
2A11	2430	
2A11	2430	192 T.POS = T.POS + 2 * N.DIAG.COL
2A20	2430	NOX = CC1(PNUM)+CC2(PNUM)+CC3(PNUM)+CC4(PNUM)+NSOX(PNUM)
2A3F	2432	
2A3F	2432	REM *** SECONDARY DX ***

Offset	Data	Source Line
2A3F	2432	INP.STO\$=MID\$(TEXT\$,T.POS,NDX)
2A53	2432	DX2.BUF\$=""
2A5C	2436	N.STO=1
2A63	2436	
2A63	2436	194 X.STO=INSTR(N.STO,INP.STO\$,"1")
2A75	2436	IF X.STO=0 THEN GOTO 195 'THATS ALL
2A84	2436	N.STO=X.STO + 1 'NEXT STARTING POINT
2A8C	2436	DX2.BUF\$=DX2.BUF\$ + DIAGN.TAB\$(X.STO) 'ADD CODE TO BUFFER BY FIVES
2AA1	2436	IF N.STO <= LEN(INP.STO\$) THEN GOTO 194
2AB4	2436	195 IF SECDX\$="" " THEN GOTO 196
2AC6	2436	DX2.BUF\$=DX2.BUF\$+SECDX\$ 'ADD OTHER SEC DX IF THERE
2AD2	2436	
2AD2	2436	196 T.POS = T.POS + NDX
2ADD	2436	
2ADD	2436	IF PGMID\$ <> "740" THEN 197 'SKIP IF NOT PSYCHOLOGY
2AEB	2436	REM *** PSYCHOMETRIC ASSESSMENTS ***
2AEB	2436	INP.STO\$=MID\$(TEXT\$,T.POS,4)
2AFE	2436	GOSUB 5700 'CONVERT ARRAY
2B03	2436	PSYCHO\$=BUF.STO\$ 'UP TO 4 TWO DIGIT CODES
2B0C	243A	T.POS = T.POS + 4
2B17	243A	
2B17	243A	197 IF PGMID\$ = "730" THEN 198 'DO SPEC PROG FOR
2B25	243A	IF PGMID\$ = "740" THEN 198 ' PSYCHOLOGY, PSYCHIATRY AND
2B33	243A	IF PGMID\$ = "780" THEN 198 ' SOCIAL WORK ONLY
2B41	243A	GOTO 199
2B45	243A	REM *** SPEC PROGRAMS ***
2B45	243A	REM *** READ AS ARRAY ***
2B45	243A	198 INP.STO\$=MID\$(TEXT\$,T.POS,7)
2B58	243A	GOSUB 5700 'CONVERT ARRAY
2B50	243A	SPPROGS=BUF.STO\$ 'UP TO 7 TWO DIGIT CODES
2B66	243E	
2B66	243E	199 REM
2B67	243E	REM -----END OF MODULE RACREAD.MOD-----
2B67	243E	
2B67	243E	IF N.ERR = 0 THEN GOTO 997
2B76	243E	LPRINT "LITHO # ";LITHO\$;" ...ERRORS"
2B88	243E	FOR 1997 = 1 TO N.ERR
2B95	2440	LPRINT USING "### ";1997;
2BA1	2442	LPRINT "=> ";ED.MSG\$(1997)
2BB7	2442	NEXT 1997
2BC8	2442	LN.COUNT = LN.COUNT + N.ERR + 1
2BD4	2444	CNTRLOPT = 6
2BD8	2444	GOSUB 9010 'REJECT THE FORM
2BE0	2444	GOTO 998 'BYPASS THE DISK WRITER....
2BE4	2444	
2BE4	2444	997 REM \$INCLUDE: 'RACWRIT.MOD' REM INCLUDE THE BASE FORM DISK WRITER
2BE5	2444	REM *****
2BE5	2444	REM **** AMBULATORY CARE INFORMATION SYSTEM 10 MAR 87 ****
2BE5	2444	REM **** D R BOLLING ****
2BE5	2444	REM **** MODULE NAME : RACWRIT.MOD ****
2BE5	2444	REM **** SCANNER PROGRAM # : GENERAL FORM ****
2BE5	2444	REM ****
2BE5	2444	REM **** PURPOSE : CREATE AND WRITE THE DISK ****
2BE5	2444	REM **** RECORD FOR INPUT TO FOCUS ****

Offset	Data	Source Line
2BE5	2444	REM ****
2BE5	2444	REM ****
2BE5	2444	REM **** RESERVED LINE NUMBERS 200-299 ****
2BE5	2444	REM ****
2BE5	2444	REM BUILD THE OUTPUT RECORD
2BE5	2444	200 GOSUB 276 'BUILD THE RECORD KEY
2BEA	2444	REM ****
2BEA	2444	REM **** RECORD TYPE "1" - MAIN TRANSACTION ****
2BEA	2444	REM ****
2BEA	2444	GOSUB 278 'BUILD DATA FOR TYPE 1
2BEF	2444	RECOU\$=PGMID\$+"1"+RECKEY\$+RECOD1\$ 'TRANSACTION ID PLUS RECORD
2C09	2450	GOSUB 280
2C0E	2450	PRINT #1,RECOU\$
2C19	2450	REM ****
2C19	2450	REM **** RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE ****
2C19	2450	REM ****
2C19	2450	REM *** ADD ADDITIONAL PROCEDURES IF ANY ***
2C19	2450	IF ADDP1\$<>" " AND PR2PRC1\$="N" THEN ESP.BUF1\$=ESP.BUF1\$+ADDP1\$
2C4B	2450	IF ADDP2\$<>" " AND PR2PRC2\$="N" THEN ESP.BUF1\$=ESP.BUF1\$+ADDP2\$
2C7D	2450	IF ADDP1\$<>" " AND PR2PRC1\$="Y" THEN ESP.BUF2\$=ESP.BUF2\$+ADDP1\$
2CAF	2450	IF ADDP2\$<>" " AND PR2PRC2\$="Y" THEN ESP.BUF2\$=ESP.BUF2\$+ADDP2\$
2CE1	2450	REM *** PROCESS PROV 1 PROCEDURES ***
2CE1	2450	IF LEN(ESP.BUF1\$)=0 THEN GOTO 206
2CF2	2450	RPOINT=1
2CF9	2452	202 RECOO2\$=MID\$(ESP.BUF1\$,RPOINT,5)
2D0C	2456	IF RECOO2\$=" " THEN GOTO 204
2D1E	2456	RECOU\$=PGMID\$+"2"+RECKEY\$+"1"+RECOO2\$ 'TRANSACTION ID PLUS RECORD
2D3E	2456	GOSUB 280
2D43	2456	PRINT #1,RECOU\$
2D4E	2456	204 RPOINT = RPOINT + 5
2D58	2456	IF RPOINT < LEN(ESP.BUF1\$) THEN GOTO 202
2D68	2456	REM *** PROCESS PROV 2 PROCEDURES ***
2D68	2456	206 IF LEN(ESP.BUF2\$)=0 THEN GOTO 212
2D7C	2456	RPOINT=1
2D83	2456	208 RECOO2\$=MID\$(ESP.BUF2\$,RPOINT,5)
2D96	2456	IF RECOO2\$=" " THEN GOTO 210
2DA8	2456	RECOU\$=PGMID\$+"2"+RECKEY\$+"2"+RECOO2\$ 'TRANSACTION ID PLUS RECORD
2DCB	2456	GOSUB 280

RACPGEN
GENERAL FORM DESTRING/DECODE PROGRAM

PAGE 20
07-16-87
14:51:36

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
2000	2456	PRINT #1,RECOU\$
2008	2456	.
2008	2456	210 RPOINT = RPOINT + 5
20E5	2456	IF RPOINT < LEN(ESP.BUF2\$) THEN GOTO 208
20F8	2456	
20F8	2456	212 REM END OF TYPE 2 RECORDS
20F9	2456	REM *****
20F9	2456	REM **** RECORD TYPE "3" - RECKEY PLUS SPECIFIC PRE CLINIC CODES ****
20F9	2456	REM *****
20F9	2456	IF LEN(SPE.BUF\$)=0 THEN 218
2E06	2456	
2E06	2456	RPOINT=1
2E0D	2456	214 X3\$=MID\$(SPE.BUF\$,RPOINT,2)
2E20	245A	IF X3\$="10" THEN RECOO3\$="M" :GOTO 215
2E38	245E	IF X3\$="11" THEN RECOO3\$="L" :GOTO 215
2E56	245E	IF X3\$="12" THEN RECOO3\$="X" :GOTO 215
2E71	245E	
2E71	245E	RECOO3\$=RIGHT\$(X3\$,1)
2E80	245E	IF RECOO3\$=" " THEN 216
2E8E	245E	
2E8E	245E	215 RECOU\$=PGMID\$ + "3" + RECKEY\$ + RECOO3\$
2EA8	245E	GOSUB 280
2EAD	245E	PRINT #1,RECOU\$
2EB8	245E	
2EB8	245E	216 RPOINT=RPOINT+2
2EC1	245E	IF RPOINT < LEN(SPE.BUF\$) THEN 214
2ED0	245E	
2ED0	245E	218 REM END OF TYPE 3 RECORDS
2ED1	245E	
2ED1	245E	REM *****
2ED1	245E	REM **** RECORD TYPE "4" - RECKEY PLUS OTHER CODES ****
2ED1	245E	REM *****
2ED1	245E	
2ED1	245E	IF LEN(OTH.BUF\$)=0 THEN GOTO 224
2EE2	245E	
2EE2	245E	RPOINT=1
2EE9	245E	220 RECOO4\$=MID\$(OTH.BUF\$,RPOINT,5)
2EFC	2462	IF RECOO4\$=" " THEN GOTO 222
2F0E	2462	
2F0E	2462	RECOU\$=PGMID\$+"4"+RECKEY\$+RECOO4\$ 'TRANSACTION ID PLUS RECORD
2F28	2462	GOSUB 280
2F2D	2462	PRINT #1,RECOU\$
2F38	2462	
2F38	2462	222 RPOINT = RPOINT + 5
2F42	2462	IF RPOINT < LEN(OTH.BUF\$) THEN GOTO 220
2F55	2462	
2F55	2462	
2F55	2462	224 REM END OF TYPE 4 RECORDS
2F56	2462	
2F56	2462	REM *****
2F56	2462	REM **** RECORD TYPE "5" - RECKEY PLUS GROUP 1 DATA ****
2F56	2462	REM *****
2F56	2462	
2F56	2462	REM *****

Offset	Data	Source Line
2F56	2462	REM **** RECORD TYPE "6" - RECKEY PLUS SPECIAL PROGRAMS ****
2F56	2462	REM *****
2F56	2462	IF LEN(SPPROG\$)=0 THEN 262
2F63	2462	
2F63	2462	RPOINT=1
2F6A	2462	258 RECOD6\$=RIGHT\$(MID\$(SPPROG\$,RPOINT,2),1)
2F83	2466	IF RECOD6\$=" " THEN 260
2F91	2466	
2F91	2466	RECOUT\$=PGMID\$ + "6" + RECKEY\$ + RECOD6\$
2FAB	2466	GOSUB 280
2FB0	2466	PRINT #1,RECOUT\$
2FBB	2466	
2FB8	2466	260 RPOINT=RPOINT+2
2FC4	2466	IF RPOINT < LEN(SPPROG\$) THEN 258
2FD0	2466	
2FD0	2466	262 REM END OF TYPE 6 RECORDS
2FD1	2466	
2FD1	2466	REM *****
2FD1	2466	REM **** RECORD TYPE "7" - RECKEY PLUS SECOND DX CODE ****
2FD1	2466	REM *****
2FD1	2466	
2FD1	2466	IF LEN(DX2.BUF\$)=0 THEN GOTO 268
2FE2	2466	
2FE2	2466	RPOINT=1
2FE9	2466	264 RECOD7\$=MID\$(DX2.BUF\$,RPOINT,5)
2FFC	246A	IF RECOD7\$=" " THEN GOTO 266
300E	246A	
300E	246A	RECOUT\$ =PGMID\$+"7"+RECKEY\$+RECOD7\$ 'TRANSACTION ID 7
3028	246A	GOSUB 280
302D	246A	PRINT #1,RECOUT\$
3038	246A	
3038	246A	266 RPOINT = RPOINT + 5
3042	246A	IF RPOINT < LEN(DX2.BUF\$) THEN GOTO 264
3055	246A	
3055	246A	268 REM END OF TYPE 7 RECORDS
3056	246A	
3056	246A	REM *****
3056	246A	REM **** RECORD TYPE "8" - RECKEY PLUS PSYCHOMETRIC ASSESSMENTS ****
3056	246A	REM *****
3056	246A	IF LEN(PSYCHOS\$)=0 THEN 274
3063	246A	
3063	246A	RPOINT=1
306A	246A	270 RECOD8\$=RIGHT\$(MID\$(PSYCHOS\$,RPOINT,2),1)
3083	246E	IF RECOD8\$=" " THEN 272
3091	246E	
3091	246E	RECOUT\$=PGMID\$ + "8" + RECKEY\$ + RECOD8\$
30AB	246E	GOSUB 280
30B0	246E	PRINT #1,RECOUT\$
30B8	246E	
30B8	246E	272 RPOINT=RPOINT+2
30C4	246E	IF RPOINT < LEN(PSYCHOS\$) THEN 270
30D0	246E	
30D0	246E	274 REM END OF TYPE 8 RECORDS
30D1	246E	

Offset	Data	Source Line
3001	246E	REM *****
3001	246E	REM PHYSICAL THERAPY MAY NEED SEPARATE RECORDS IF THE OTHER
3001	246E	REM UCA CODE IS BUBBLED. INDICATOR PHTH=1 IF CODED.
3001	246E	REM *****
3001	246E	IF PHTH<>1 THEN 275 ' GO IF NOT =1
300C	246E	PHTH=0 ' RESET INDICATOR
30E3	246E	CL1.COD\$="DHDA" ' PROVIDE NEW UCA CODE
30EC	246E	ADDP1\$="" :ADDP2\$="" ' DONT ADD THESE AGAIN THE SECOND TIME AROUND
30FE	246E	GOTO 200 ' GO AROUND AGAIN
3102	246E	
3102	246E	275 GOTO 299
3106	246E	
3106	246E	REM *****
3106	246E	**** SUBROUTINE 270 - BUILD THE RECORD KEY ****
3106	246E	REM *****
3106	246E	276 RECKEY\$=""
310F	246E	
310F	246E	REM *** CLINIC ID (PREFIX + COD) ***
310F	246E	RECKEY\$ = CL1.COD\$
3118	246E	
3118	246E	REM *** VISIT DATE ***
3118	246E	RECKEY\$ = RECKEY\$ + VIDATES
3124	246E	
3124	246E	REM *** PRIMARY PROVIDER ***
3124	246E	RECKEY\$ = RECKEY\$ + PPROV.PFX\$ + PPROV.NUM\$
3137	246E	
3137	246E	REM *** PATIENT SSN ***
3137	246E	RECKEY\$ = RECKEY\$ + SSN\$
3143	246E	
3143	246E	REM *** FAMILY MEMBER PREF ***
3143	246E	RECKEY\$ = RECKEY\$ + FMEMP\$
314F	246E	
314F	246E	REM *** LITHO CODE ***
314F	246E	RECKEY\$ = RECKEY\$ + LITHOS
315B	246E	
315B	246E	REM *** FORM NUMBER ***
315B	246E	RECKEY\$ = RECKEY\$ + FRMNS
3167	246E	
3167	246E	RETURN
316A	246E	
316A	246E	REM *****
316A	246E	**** SUBROUTINE 272 - BUILD THE DATA FOR TYPE 1 ****
316A	246E	REM *****
316A	246E	278 RECOD1\$=""
3173	246E	
3173	246E	REM *** VISIT COUNT ***
3173	246E	RECOD1\$ = RECOD1\$ + VCNT\$
317F	246E	
317F	246E	REM *** PRIMARY PROV TIME ***
317F	246E	RECOD1\$ = RECOD1\$ + PPROV.TIM\$
3188	246E	
3188	246E	REM *** SECONDARY PROVIDER ***
3188	246E	RECOD1\$ = RECOD1\$ + SPROV.PFX\$ + SPROV.NUM\$
319E	246E	

Offset	Data	Source Line
319E	246E	REM *** SECONDARY PROVIDER TIME ***
319E	246E	RECOD1\$ = RECOD1\$ + SPROV.TIM\$
31AA	246E	
31AA	246E	REM *** REASON FOR SECONDARY PROVIDER ***
31AA	246E	RECOD1\$ = RECOD1\$ + SPROV.REAS
31B6	246E	
31B6	246E	REM *** APPORINTMENT STATUS ***
31B6	246E	RECOD1\$ = RECOD1\$ + NOTSCS
31C2	246E	
31C2	246E	REM *** REFERRAL CODE ***
31C2	246E	RECOD1\$ = RECOD1\$ + REF.PFX\$ + REF.COD\$
31D5	246E	
31D5	246E	REM *** PLACE OF VISIT ***
31D5	246E	RECOD1\$ = RECOD1\$ + NOTCL\$
31E1	246E	
31E1	246E	REM *** JOB RELATED VISIT ***
31E1	246E	RECOD1\$ = RECOD1\$ + RELAT.VISS
31ED	246E	
31ED	246E	REM *** MIL ONLY DUTY ***
31ED	246E	RECOD1\$ = RECOD1\$ + MILDUT\$
31F9	246E	
31F9	246E	REM *** MIL ONLY QTRS ***
31F9	246E	RECOD1\$ = RECOD1\$ + MILQTRS
3205	246E	
3205	246E	REM *** MIL ONLY PROFILE ***
3205	246E	RECOD1\$ = RECOD1\$ + MILPRO\$
3211	246E	
3211	246E	REM *** NOT AVAILABLE ***
3211	246E	RECOD1\$ = RECOD1\$ + NAVAIL\$
321D	246E	
321D	246E	REM *** ADMITTED ***
321D	246E	RECOD1\$ = RECOD1\$ + ADMITS
3229	246E	
3229	246E	REM *** INFIELD (NOT ON GENERAL FORM) ***
3229	246E	RECOD1\$ = RECOD1\$ + " "
3235	246E	
3235	246E	REM *** INJURY (NOT ON GENERAL FORM) ***
3235	246E	RECOD1\$ = RECOD1\$ + " "
3241	246E	
3241	246E	REM *** PURPOSE OF VISIT (NOT ON GENERAL FORM) ***
3241	246E	RECOD1\$ = RECOD1\$ + " "
324D	246E	
324D	246E	REM *** PRIM FOLLOW-UP/RULE OUT ***
324D	246E	RECOD1\$ = RECOD1\$ + FU.ROS
3259	246E	
3259	246E	REM *** PRIMARY DX CODE ***
3259	246E	RECOD1\$ = RECOD1\$ + PRIMDX\$
3265	246E	
3265	246E	RETURN
3268	246E	
3268	246E	REM *****
3268	246E	REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
3268	246E	REM *****
3268	246E	280 PAD=MAXLENGTH - LEN(RECOUT\$) 'FIND OUT HOW SHORT THE RECORD IS

RACPGEN
GENERAL FORM DESTSTRING/DECODE PROGRAM

PAGE 24

07-06-87

14:51:36

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
3276	2470	RECOU\$ = RECOU\$ + STRING\$(PAD,PAD\$) 'PAD THE RECORD WITH FILL CHAR
3289	2470	RETURN
328C	2470	
328C	2470	299 REM
328D	2470	
328D	2470	REM -----END OF MODULE RACWRIT.MOD-----
328D	2470	
328D	2470	998 REM CONTINUE
328E	2470	
328E	2470	999 READTYPE = 2
3295	2470	IF LNLCOUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
32A5	2472	GOTO 10
32A9	2472	
32A9	2472	REM END OF SCAN/DECODE/WRITE LOOP =====
32A9	2472	
32A9	2472	1000 REM \$INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
32AA	2472	REM =====
32AA	2472	REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
32AA	2472	REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
32AA	2472	REM =====
32AA	2472	REM PATIENT OMR INPUT PROGRAM *
32AA	2472	REM *
32AA	2472	REM This program verifies user is logged on properly. If there is no *
32AA	2472	REM valid user logged on at the time of execution, this subroutine will*
32AA	2472	REM chain to the logon program RACP05, otherwise a return is issued. *
32AA	2472	REM =====
32AA	2472	REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
32AA	2472	REM =====
32AA	2472	1001 OPEN "I",1,"RACLOG.DAT"
32BC	2472	IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
32CA	2472	INPUT #1,USERS(1),DT\$,TMS\$,PID\$
32EB	247E	IF USERS(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
32F9	247E	IF USERS(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
3307	247E	CLOSE 1
330E	247E	SCREEN 0,1,0,0
3324	247E	COLOR FORE,BACK,BORD
333A	247E	CLS
333E	247E	RETURN
3341	247E	
3341	247E	1002 CLOSE
3345	247E	CHAIN "RACP05"
334C	247E	'=====END OF LOGON VERIFY SUBROUTINE 1000=====
334C	247E	
334C	247E	2000 REM \$INCLUDE: 'RACS2000.SUB' INCLUDE THE RIPLY/DELAY SUB
334D	247E	REM =====
334D	247E	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
334D	247E	REM **** SKIP COLE ****
334D	247E	REM **** SUBROUTINE NAME : RACS2000.SUB ****
334D	247E	REM **** SCANNER PROGRAM # : ALL ****
334D	247E	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
334D	247E	REM **** SERVERS AS A WAIT AND REPLY ****
334D	247E	REM **** ENTRY MODULE ****
334D	247E	REM **** INPUT : SINGLE KEYBOARD ENTRY ****
334D	247E	REM ****

IBM Personal Computer BASIC Compiler V1.00

```

Offset  Data  Source Line
334D 247E REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
334D 247E REM ****
334D 247E REM **** RESERVED LINE ****
334D 247E REM **** NUMBERS : 2001-2010 ****
334D 247E REM *****
334D 247E 2001 REM REPLY FUNCTION
334E 247E 2002 REPLY$=INKEY$ : IF REPLY$="" THEN 2002
3362 2482 REPLY=ASC(REPLY$)
336C 2484 IF REPLY > 90 THEN REPLY$=CHR$(REPLY XOR 32) 'CONVERT TO CAPS
3387 2484 IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$="?"
3383 2484 RETURN
3386 2484
3386 2484 5000 REM $INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB
3387 2484 REM *****
3387 2484 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
3387 2484 REM **** SKIP COLE ****
3387 2484 REM **** SUBROUTINE NAME : RXXS5000.SUB ****
3387 2484 R M **** SCANNER PROGRAM # : ALL ****
3387 2484 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
3387 2484 REM **** PERFORMS A DATE EDIT ****
3387 2484 REM ****
3387 2484 REM **** INPUT : DATE TO BE CHECKED MUST BE ****
3387 2484 REM **** IN THE VARIABLE NAMED ****
3387 2484 REM **** 'CK.5000$' ****
3387 2484 REM **** IN THE FORMAT "YYMMDD" ****
3387 2484 REM ****
3387 2484 REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
3387 2484 REM **** VARIABLE. IF THIS VARIABLE ****
3387 2484 REM **** CONTAINS ANY NUMBER OTHER ****
3387 2484 REM **** THAN 0, AN ERROR WAS FOUND ****
3387 2484 REM **** IN THE DATE. ****
3387 2484 REM ****
3387 2484 REM **** RESERVED LINE ****
3387 2484 REM **** NUMBERS : 5001-5009 ****
3387 2484 REM *****
3387 2484 RT.5000 = 0
338E 2484 CKYEAR = VAL(LEFT$(CK.5000$,2)) 'YEAR NUMERIC VALUE
33D1 2486 CKMONTH = VAL(MID$(CK.5000$,3,2)) 'MONTH NUMERIC VALUE
33E7 2488 CKDAY = VAL(RIGHT$(CK.5000$,2)) 'DAY NUMERIC VALUE
33FA 248A
33FA 248A IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
3410 248A IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
3426 248A IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
343C 248A IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
3452 248A
3452 248A REM LEAP YEAR CHECK
3452 248A MOLENGTH(2) = 28
3459 248A IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY
3468 248A IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR
347D 248A MOLENGTH(2) = 29
3484 248A
3484 248A 5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
34A3 248A
34A3 248A 5009 RETURN

```

Offset	Data	Source Line
34A6	248A	
34A6	248A	REM -----END OF SUBROUTINE 5000 -----
34A6	248A	
34A6	248A	
34A6	248A	5500 REM INCLUDE: 'RACSS500.SUB' INCLUDE THE OUTP UCA VALIDATE SUB
34A7	248A	5600 REM INCLUDE: 'RACSS600.SUB' INCLUDE THE INP UCA VALIDATE SUB
34A8	248A	5700 REM \$INCLUDE: 'RACSS700.SUB' INCLUDE MAP ONES TO POSITION NO.
34A9	248A	REM *****
34A9	248A	REM **** AMBULATORY CARE DATA BASE 29 JUL 85 ****
34A9	248A	REM **** D R BOLLING ****
34A9	248A	REM **** SUBROUTINE NAME : RXXS5700.SUB ****
34A9	248A	REM **** SCANNER PROGRAM # : ALL ****
34A9	248A	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
34A9	248A	REM **** CONVERTS A BINARY ARRAY INTO ****
34A9	248A	REM **** TWO CHAR CODES. ****
34A9	248A	REM ****
34A9	248A	REM **** INPUT : INP.STO\$ AS STRING ****
34A9	248A	REM ****
34A9	248A	REM ****
34A9	248A	REM **** OUTPUT : BUF.STO\$ AS STRING. ****
34A9	248A	REM ****
34A9	248A	REM ****
34A9	248A	REM **** RESERVED LINE ****
34A9	248A	REM **** NUMBERS : 5710-5730 ****
34A9	248A	REM *****
34A9	248A	
34A9	248A	BUF.STO\$=""
34B2	248A	N.STO=1
34B9	248A	5710 X.STO=INSTR(N.STO,INP.STO\$,"1")
34CB	248A	IF X.STO=0 THEN GOTO 5720 'THATS ALL
34DA	248A	N.STO = X.STO + 1 'NEXT STARTING POINT
34E2	248A	X.STO = X.STO + 100 'PAD WITH LEADING ZERO
34EC	248A	BUF.STO\$ = BUF.STO\$ + RIGHT\$(STR\$(X.STO),2)
3502	248A	IF N.STO <= LEN(INP.STO\$) THEN GOTO 5710
3515	248A	
3515	248A	5720 RETURN
3518	248A	
3518	248A	REM ----- END OF SUBROUTINE 5700 -----
3518	248A	5800 REM \$INCLUDE: 'RACSS800.SUB' INCLUDE FOUR COL DX CONVERTER
3519	248A	REM *****
3519	248A	REM *** AMBULATORY CARE DATA BASE 30 JUL 85 ***
3519	248A	REM *** D R BOLLING ***
3519	248A	REM *** SUBROUTINE NAME : RXXS5800.SUB ***
3519	248A	REM *** SCANNER PROGRAM : AS APPROPRIATE ***
3519	248A	REM *** FUNCTION : THIS SUBROUTINE MODULE READS ***
3519	248A	REM *** FOUR COLUMNS IN DX AREA AND ***
3519	248A	REM *** CONVERTS TO A POSITION IN A ***
3519	248A	REM *** TABLE. AN ERROR CODE IS ***
3519	248A	REM *** RETURNED IF MULTIPLE CODES ARE ***
3519	248A	REM *** PRESENT. ***
3519	248A	REM ***
3519	248A	REM *** INPUT : X.POS - STARTING POSITION IN ***
3519	248A	REM *** STRING ***
3519	248A	REM ***

Offset	Data	Source Line
3519	248A	REM *** OUTPUT : X.FIN - TABLE POSITION OF DX ***
3519	248A	REM *** RT.5800 = 1 ON ERROR ***
3519	248A	REM ***
3519	248A	REM *** RESERVED LINE NUMBERS : 5801 - 5899 ***
3519	248A	REM *****
3519	248A	RT.5800=0 'INITIALIZE ERROR INDICATOR
3520	248A	X.FIN=0 'INITIALIZE TABLE OFFSET RESULT
3527	248A	
3527	248A	REM ** COLUMN 1 **
3527	248A	X=0 'STARTING POINTER
352E	248A	X.SIZ=C1SIZ
3535	248C	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
354C	248C	IF X.STO=0 THEN GOTO 5802
355B	248C	X.FIN=X.STO
3562	248C	
3562	248C	5802 REM ** COLUMN 2 **
3563	248C	IF C2SIZ=0 THEN GOTO 5804
3572	248C	X=X+X.SIZ
357D	248C	X.POS=X.POS+2
3586	248C	X.SIZ=C2SIZ
358D	248C	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
35A4	248C	IF X.STO=0 THEN GOTO 5804
3583	248C	IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
35C2	248C	X.FIN=X.STO + X
35CD	248C	
35CD	248C	5804 REM ** COLUMN 3 **
35CE	248C	IF C3SIZ=0 THEN GOTO 5806
35D0	248C	X=X+X.SIZ
35E8	248C	X.POS=X.POS+2
35F1	248C	X.SIZ=C3SIZ
35F8	248C	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
360F	248C	IF X.STO=0 THEN GOTO 5806
361E	248C	IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
362D	248C	X.FIN=X.STO + X
3638	248C	
3638	248C	5806 REM ** COLUMN 4 **
3639	248C	IF C4SIZ=0 THEN GOTO 5808
3648	248C	X=X+X.SIZ
3653	248C	X.POS=X.POS+2
365C	248C	X.SIZ=C4SIZ
3663	248C	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
367A	248C	IF X.STO=0 THEN GOTO 5808
3689	248C	IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
3698	248C	X.FIN=X.STO + X
36A3	248C	
36A3	248C	5808 REM
36A4	248C	GOTO 5899
36A8	248C	
36A8	248C	5890 RT.5800=1 'ERROR - MULTIPLE CODES
36AF	248C	
36AF	248C	5899 RETURN
36B2	248C	
36C2	248C	REM -----END OF SUBROUTINE RXXS5800.SUB-----
36B2	248C	

Offset	Data	Source Line
3682	248C	
3682	248C	6000 REM \$INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE SUB
3683	248C	REM *****
3683	248C	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
3683	248C	REM **** SKIP COLE ****
3683	248C	REM **** SUBROUTINE NAME : RXXS6000.SUB ****
3683	248C	REM **** SCANNER PROGRAM # : ALL ****
3683	248C	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
3683	248C	REM **** PERFORMS INSTRING SEARCH ****
3683	248C	REM ****
3683	248C	REM **** INPUT : STRING TO BE SEARCHED MUST ****
3683	248C	REM **** BE IN THE VARIABLE NAMED : ****
3683	248C	REM **** 'X\$' ****
3683	248C	REM ****
3683	248C	REM **** OUTPUT : 'TOT' = TOTAL NUMBER OF ****
3683	248C	REM **** HITS IN THE DESTING ****
3683	248C	REM **** 'HOLD\$()' IS THE ARRAY ****
3683	248C	REM **** CONTAINING THE NUMERIC ****
3683	248C	REM **** VALUE OF THE HIT POSITIONS ****
3683	248C	REM ****
3683	248C	REM *****
3683	248C	REM **** RESERVED LINE ****
3683	248C	REM **** NUMBERS : 6001-6009 ****
3683	248C	REM *****
3683	248C	6001 PTR = INSTR(X\$, "1")
36C1	248E	TOT = 0
36C8	2490	WHILE PTR > 0
36D3	2490	TOT=TOT+1
36D8	2490	HOLD\$(TOT) = RIGHT\$(STR\$(PTR),2)
36FD	2490	PTR=PTR+1
3705	2490	PTR = INSTR(PTR,X\$, "1")
3717	2490	WEND
3718	2490	RETURN
371E	2490	
371E	2490	REM -----END OF SUBROUTINE RXXS6000.SUB-----
371E	2490	
371E	2490	
371E	2490	7000 REM \$INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
371F	2490	REM *****
371F	2490	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
371F	2490	REM **** SKIP COLE ****
371F	2490	REM **** SUBROUTINE NAME : RACS7000.SUB ****
371F	2490	REM **** SCANNER PROGRAM # : ALL ****
371F	2490	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
371F	2490	REM **** PRINTS THE STANDARD SCREEN ****
371F	2490	REM **** HEADING. ****
371F	2490	REM **** INPUT : COMMON VARIABLE USER\$(2) ****
371F	2490	REM **** SYSTEM DATE ****
371F	2490	REM ****
371F	2490	REM **** OUTPUT : SCREEN HEADING ****
371F	2490	REM ****
371F	2490	REM **** RESERVED LINE ****
371F	2490	REM **** NUMBERS : 7001-7010 ****
371F	2490	REM *****

Offset	Data	Source Line
371F	2490	
371F	2490	7001 LOCATE 1,1
3729	2490	PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
3731	2490	LOCATE 1,65
373E	2490	PRINT DATES;
3746	2490	LOCATE 2,1
3753	2490	PRINT "USER : ";USERS(1)
3760	2490	RETURN
3763	2490	
3763	2490	7100 REM \$INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
3764	2490	REM *****
3764	2490	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
3764	2490	REM **** SKIP COLE ****
3764	2490	REM **** SUBROUTINE NAME : RXXS7100.SUB ****
3764	2490	REM **** SCANNER PROGRAM # : ALL ****
3764	2490	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
3764	2490	REM **** PRINTS THE STANDARD HEADING ****
3764	2490	REM **** ON THE PRINTER. ****
3764	2490	REM **** INPUT : DATE,PAGE,PGMID\$,PGMTITL\$ ****
3764	2490	REM **** ****
3764	2490	REM **** OUTPUT : PRINTER HEADING, LN.COUNT ****
3764	2490	REM **** ****
3764	2490	REM **** RESERVED LINE ****
3764	2490	REM **** NUMBERS : 7101-7110 ****
3764	2490	REM *****
3764	2490	
3764	2490	7101 IF PAGE > 0 THEN LPRINT CHR\$(12);
377A	2490	LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITL\$;
3787	2494	LPRINT TAB(70);DATES
379A	2494	PAGE=PAGE+1
37A2	2494	LPRINT "PROGRAM ";PGMID\$;TAB(70);"PAGE";
37BF	2494	LPRINT USING "####";PAGE
37CB	2494	LPRINT
37D3	2494	LN.COUNT=3
37DA	2494	RETURN
37DD	2494	
37DD	2494	8000 REM \$INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
37DE	2494	REM *****
37DE	2494	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ***
37DE	2494	REM **** SKIP COLE ***
37DE	2494	REM **** SUBROUTINE NAME : RXXS8000.SUB ***
37DE	2494	REM **** SCANNER PROGRAM # : ALL ***
37DE	2494	REM **** FUNCTION : THIS SUBROUTINE MODULE ***
37DE	2494	REM **** IS A GROUPING THAT PERFORMS ***
37DE	2494	REM **** VARIOUS DECODING FUNCTIONS ***
37DE	2494	REM **** ON THE SCANNER DATA ***
37DE	2494	REM **** ****
37DE	2494	REM **** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
37DE	2494	REM **** 8050 - CHECK FOR END OF JOB ***
37DE	2494	REM **** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
37DE	2494	REM **** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
37DE	2494	REM **** (RETURNED IN TEXT\$ STRING VARIABLE) ***
37DE	2494	REM **** ****
37DE	2494	REM **** INPUT : SHEET RECORD, RECORD LENGTH ***

RACPGEN

GENERAL FORM DESTRING/DECODE PROGRAM

PAGE 30

07-06-87

14:51.36

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

37DE 2494 REM **** ***
37DE 2494 REM **** OUTPUT : 'TEXT$' TRING VARIABLE ***
37DE 2494 REM **** ***
37DE 2494 REM **** RESERVED LINE ***
37DE 2494 REM **** NUMBERS : 8001-8500 ***
37DE 2494 REM *****
37DE 2494
37DE 2494 'DECODE THE HEADER ONLY
37DE 2494 8001 POINTER = 0
37E5 2496 RECORDPTR = VARPTR(SHEETREC(0))
37EC 2498 FOR J8000 = 1 TO 21
37F3 2498 8002 TEXT$= TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
3811 2498 POINTER=POINTER+1
3819 2498 NEXT J8000
3828 249A PROGRAM$= LEFT$(TEXT$,3)
3837 249E BATCH$= MID$(TEXT$,4,3)
3849 24A2 SERIAL$= MID$(TEXT$,7,4)
385B 24A6 RUNID$= MID$(TEXT$,11,1)
386D 24AA FORM$= MID$(TEXT$,12,2)
387F 24AE POCKET$= MID$(TEXT$,14,1)
3891 24B2 SCANERR1$=MID$(TEXT$,16,2)
38A3 24B6 SCANERR2$=MID$(TEXT$,18,2)
38B5 24BA SCANERR3$=MID$(TEXT$,20,2)
38C7 24BE GOTO 8500
38C8 24BE
38C8 24BE 8050 REM CHECK FOR END OF JOB/END OF BATCH
38CC 24BE IF PROGRAM$ = PGMD$ THEN GOTO 8500
38DE 24BE LPRINT STRING$(80,"")
38EC 24BE LPRINT
38F4 24BE LPRINT "RECORDS PROCESSED ... ";SERIAL$
3901 24BE LPRINT "STARTED AT ..... ";BTIM$
390E 24BE LPRINT "ENDED AT ..... ";TIM$
391B 24BE LPRINT CHR$(12)
3926 24BE GOTO 30000
392A 24BE
392A 24BE 8070 REM CHECK FOR SCANNER ERRORS
392B 24BE IF POCKET$ = " " GOTO 8500
3930 24BE LPRINT LITHOS;
3945 24BE LPRINT " ... SCANNER ERRORS : ";
394D 24BE LPRINT SCANERR1$;" / ";
395A 24BE LPRINT SCANERR2$;" / ";
3967 24BE LPRINT SCANERR3$
396F 24BE LN=LN+1
3977 24C0 GOTO 999
397B 24C0
397B 24C0 8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
397C 24C0 LOCATE 5,1:PRINT "PROGRAM ";PROGRAM$;
3996 24C0
3996 24C0 PRINT " BATCH ";BATCH$;
39A3 24C0 PRINT " RUN ";RUNID$;
39B0 24C0 PRINT " FORM ";FORM$;
39B0 24C0 PRINT " POCKET ";POCKET$
39CA 24C0 GOTO 8500
39CE 24C0

```

```

Offset  Data  Source Line
39CE 24C0 8200 REM DECODE THE RESPONSE POSITIONS
39CF 24C0      POINTER = 21
39D6 24C0      RECORDPTR = VARPTR(SHEETREC(0))
39D0 24C0      FOR J8000 = 22 TO RECORDLENGTH
39EA 24C2 8202      TEXT$ = TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
3A08 24C2      POINTER=POINTER+1
3A10 24C2      NEXT J8000
3A21 24C2 8500 RETURN
3A24 24C2
3A24 24C2      REM ----- END OF RXXS8000.SUB -----
3A24 24C2
3A24 24C2 9000 REM $INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
3A25 24C2 REM *****
3A25 24C2 REM ****      AMBULATORY CARE DATA BASE      13 APR 85      ****
3A25 24C2 REM ****      SKIP COLE      ****
3A25 24C2 REM ****      PROGRAM NAME      :      RACS9000.SUB      ****
3A25 24C2 REM ****      SCANNER PROGRAM #      :      ALL      ****
3A25 24C2 REM ****      FUNCTION      :      THIS SUBROUTINE MODULE      ****
3A25 24C2 REM ****      CONTROLS THE SCANNER I/O      ****
3A25 24C2 REM ****
3A25 24C2 REM ****      INPUT/OUTPUT      :      REFER TO THE ASYNCHRONOUS      ****
3A25 24C2 REM ****      COMMUNICATIONS MANUAL AND THE      ****
3A25 24C2 REM ****      PRE-RELEASED SOFTWARE GUIDE      ****
3A25 24C2 REM ****
3A25 24C2 REM *****
3A25 24C2 REM ****      RESERVED LINE      ****
3A25 24C2 REM ****      NUMBERS      :      9001-9100      ****
3A25 24C2 REM *****
3A25 24C2
3A25 24C2 REM *****
3A25 24C2 REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER      ****
3A25 24C2 REM **** ARGUMENTS: PRESET ... SEE BELOW      ****
3A25 24C2 REM *****
3A25 24C2 9001 REM
3A26 24C2      PROTOCOL(0) = 9600      'BAUD RATE
3A2D 24C2      PROTOCOL(1) = 78      'PARITY (SEE PAGE 4-8 OF MANUAL)
3A34 24C2      PROTOCOL(2) = 8      'DATA BITS
3A38 24C2      PROTOCOL(3) = 1      'STOP BITS
3A42 24C2      PROTOCOL(4) = 2      'RS-232 PORT
3A49 24C2      PROTOCOL(5) = 0      'WRITE TIME-OUT
3A50 24C2      PROTOCOL(6) = 0      'READ TIME-OUT
3A57 24C2
3A57 24C2      ERRSTAT$ = SPACES(60)
3A63 24C2      ARGPTR = VARPTR(PROTOCOL(0))
3A6A 24C4      CALL SETUP (ARGPTR,ERRSTAT$)
3A78 24C4      ERRMSG$=""
3A84 24C4      IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="SETUP ERROR "+ERRSTAT$
3AA0 24C4      GOTO 9100
3AA4 24C4
3AA4 24C4 REM *****
3AA4 24C4 REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER      ****
3AA4 24C4 REM **** ARGUMENTS: CNTRLOPT      ****

```

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
3AA4	24C4	REM **** CNTRLOPT = 1 = START SCANNER (S1) *****
3AA4	24C4	REM **** CNTRLOPT = 2 = STOP SCANNER (S0) *****
3AA4	24C4	REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) *****
3AA4	24C4	REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) *****
3AA4	24C4	REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" *****
3AA4	24C4	REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" *****
3AA4	24C4	REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) *****
3AA4	24C4	REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) *****
3AA4	24C4	REM *****
3AA4	24C4	9010 REM
3AA5	24C4	ERRSTAT\$ = SPACES(60)
3AB1	24C4	CALL CNTRLOP (CNTRLOPT,ERRSTAT\$)
3AC2	24C4	ERRMSG\$=""
3AC8	24C4	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
3AE7	24C4	GOTO 9100
3AEB	24C4	REM *****
3AEB	24C4	REM **** SUBROUTINE 9020 - SCAN SHEET CALL *****
3AEB	24C4	REM **** *****
3AEB	24C4	REM **** ARGUMENTS: READTYPE *****
3AEB	24C4	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER *****
3AEB	24C4	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT *****
3AEB	24C4	REM **** *****
3AEB	24C4	REM **** ARGUMENTS: RECORDLENGTH *****
3AEB	24C4	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE *****
3AEB	24C4	REM **** TRANSMITTED *****
3AEB	24C4	REM *****
3AEB	24C4	9020 REM
3AEC	24C4	ERRSTAT\$ = SPACES(60)
3AF8	24C4	RECORDPTR = VARPTR(SHEETREC(0))
3AFF	24C4	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
3B18	24C4	ERRMSG\$=""
3B21	24C4	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
3B42	24C4	GOTO 9100
3B46	24C4	REM *****
3B46	24C4	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL *****
3B46	24C4	REM **** *****
3B46	24C4	REM **** ARGUMENTS: PRINTPOS *****
3B46	24C4	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION *****
3B46	24C4	REM **** VALUES = 0 THRU 90 *****
3B46	24C4	REM **** *****
3B46	24C4	REM **** ARGUMENTS: PSTRING\$ *****
3B46	24C4	REM **** TEXT TO BE PRINTED ON THE FORM *****
3B46	24C4	REM **** *****
3B46	24C4	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN *****
3B46	24C4	REM **** HEADER SHEET IS MARKED 'PRINTER ON' *****
3B46	24C4	REM *****
3B46	24C4	9030 REM
3B47	24C4	ERRSTAT\$ = SPACES(60)
3B53	24C4	RECORDPTR = VARPTR(SHEETREC(0))
3B5A	24C4	CALL TPRINT(PRINTPOS,PSTRING\$,ERRSTAT\$)
3B6F	24C4	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
3B8B	24C4	GOTO 9100

RACPGEN
GENERAL FORM DESTING/DECODE PROGRAM

PAGE 33

07-06-87

14:51:36

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
388F	24CA	
388F	24CA	9100 RETURN
3892	24CA	REM -----END OF SUBROUTINE RACS9000.SUB -----
3892	24CA	
3892	24CA	REM END OF SUBROUTINES =====
3892	24CA	
3892	24CA	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
3893	24CA	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
389B	24CA	LPRINT "ERASING FILE ";DATFIL\$
38A8	24CA	BEEP
38AC	24CA	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
38B8	24CA	CLOSE
38BC	24CA	OPEN DATFIL\$ FOR OUTPUT AS #1
38CE	24CA	PRINT #1,STPINGS\$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
38E0	24CA	CLOSE
38E4	24CA	
38E4	24CA	30000 REM
38E5	24CA	CLOSE
38E9	24CA	CHAIN "RACP10"
38F0	24CA	
38F3	24CA	

22151 Bytes Available
8797 Bytes Free

0 Warning Error(s)
0 Severe Error(s)

PAGE 1

05-26-87

09:33:12

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

001A 0002 REM \$LINESIZE: 132
001A 0002 REM \$PAGESIZE: 66
001A 0002 REM \$TITLE: 'RACP800 '
001A 0002 REM \$SUBTITLE: 'BAS/TMC FORM DESTING/DECODE PROGRAM'
001A 0002 REM \$PAGE

```

Offset  Data  Source Line
001A 0002 REM +-----+
001A 0002 REM | NAME: RACP800          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | DATE: 16 APR 87      BAS/TMC (BHAIE) FORM PROGRAM   |
001A 0002 REM | D R BOLLING                               |
001A 0002 REM +-----+
001A 0002 REM          BAS/TMC FORM OMR INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the base form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RACP05, otherwise,
001A 0002 REM the program chains to program RACP10 on exit.
001A 0002 REM
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD          DIMENSION DEF:   NS      *
001A 0002 '* Date: 28 Feb 84          Written by: Flu   Jole   *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINIT A-Z
001A 0002 DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002
001A 0002 DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 DIM PROTOCOL(7)    '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 DIM YN$(3)         '(YES/NO ANSWERS 0=?, 1 = Y , 2=N , 3=X)
001A 0002 DIM ED.MSG$(30)    '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002 DIM CLINIC1.PFX$(5) '(PREFIX -H J- FOR CLINIC #1 TMC)
001A 0002 DIM PROVIDER.TIME$(9) '(TIME TABLE FOR PROVIDERS)
001A 0002 DIM PROCED$(030)    '(PROCEDURE TABLE FOR BASE FORM)
001A 0002 DIM DIAGN.TAB$(160) '(DIAGNOSIS TABLE FOR BASE FORM)
001A 0002 DIM HOLD$(6000)    '(HOLD AREA FOR SUBROUTINE 6000)

```

Offset	Data	Source Line
001A	0002	DIM SPECL\$(09) '(SPECIAL PROGRAMS)
001A	0002	
001A	0002	REM \$INCLUDE: 'RACCMN.MOD' REM INCLUDE THE COMMON AREA DEFINITION
001A	0002	*****
001A	0002	* NAME: RACCMN.MOD COMMON AREA DEFINITION *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A	0002	' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	' *****START OF COMMON DEFINITIONS*****
001A	0002	
001A	0002	COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL\$ 'BASIC SCREEN COLORS
001A	0002	COMMON HEADERS\$ '21 CHARACTER SCANNER HEADER INFO
001A	0002	COMMON TEXT\$ ' AINING CHARACTERS FROM SCANNER
001A	0002	COMMON PGMID\$ 'PROGRAM OR FORM ID
001A	0002	COMMON MOLENGTH() 'DAYS IN THE MONTH
001A	0002	COMMON USER\$()
001A	0002	' *****END OF COMMON DEFINITION*****
001A	0002	
001A	0002	
001A	0002	REM \$INCLUDE: 'RACDEF.MOD' REM INCLUDE THE DEFAULT DEFINITIONS
001A	0002	*****
001A	0002	* NAME: RACP01.DEF DEFAULT DEFINITIONS *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' Variables used in common that have a default value on start*up
001A	0002	' will be held in this file. It is an included file so it cannot
001A	0002	' be run in a stand*alone mode. In normal operation, this file
001A	0002	' should be 'included' in the main program only (RACP10.BAS).
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	
001A	0002	' *****START OF DEFAULT DEFINITION*****
001A	0002	FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
0047	1354	BACK = 1 'Background Color = Light Blue
004E	1354	BORD = 4 'BORDER = RED
0055	1356	HIDE = 4 'ALTERNATE COLOR = RED
005C	1356	EFORE = 14 'ERROR FOREGROUND DISPLAY
0063	1356	EBACK = 0 'ERROR BACKGROUND DISPLAY
006A	1356	BELL\$ = CHR\$(7) 'Sound the bell
0076	1356	
0076	1356	MOLENGTH(1) = 31 'JAN
007D	1356	MOLEN.TH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
0084	1356	MOLENGTH(3) = 31 'MAR
008B	1356	MOLENGTH(4) = 30 'APR
0092	1356	MOLENGTH(5) = 31 'MAY

Offset	Data	Source Line
0099	1356	MOLENGTH(6) = 30 'JUN
00A0	1356	MOLENGTH(7) = 31 'JUL
00A7	1356	MOLENGTH(8) = 31 'AUG
00AE	1356	MOLENGTH(9) = 30 'SEP
00B5	1356	MOLENGTH(10) = 31 'OCT
00BC	1356	MOLENGTH(11) = 30 'NOV
00C3	1356	MOLENGTH(12) = 31 'DEC
00CA	1356	
00CA	1356	DATEERR\$(0) = " "
00D3	1356	DATEERR\$(1) = "INVALID MONTH"
00DC	1356	DATEERR\$(2) = "INVALID DAY "
00E5	1356	DATEERR\$(3) = "DAY TOO LARGE FOR MONTH CODED"
00EE	1356	
00EE	1356	MAXLENGTH = 80 'MAXIMUM LENGTH OF OUTPUT RECORD
00F5	1358	PADS = " " 'PAD CHARACTER FOR SHORT RECORDS
00FE	135C	
00FE	135C	*****END OF DEFAULT DEFINITION*****
00FE	135C	
00FE	135C	KEY OFF
0104	135C	
0104	135C	REM *****
0104	135C	
0104	135C	REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0104	135C	REM CLINIC #1
0104	135C	CLINIC1.PFX\$(0)=" "
010D	135C	CLINIC1.PFX\$(1)="H"
0116	135C	CLINIC1.PFX\$(2)="J"
011F	135C	
011F	135C	REM *** ENCOUNTER FORM PROVIDER TIME TABLE ***
011F	135C	
011F	135C	PROVIDER.TIMES\$(00)="000" ' NO TIME
0128	135C	PROVIDER.TIMES\$(01)="002" ' 2 MINUTES
0131	135C	PROVIDER.TIMES\$(02)="005" ' 5 MINUTES
013A	135C	PROVIDER.TIMES\$(03)="010" ' 10 MINUTES
0143	135C	PROVIDER.TIMES\$(04)="015" ' 15 MINUTES
014C	135C	PROVIDER.TIMES\$(05)="020" ' 20 MINUTES
0155	135C	PROVIDER.TIMES\$(06)="030" ' 30 MINUTES
015E	135C	PROVIDER.TIMES\$(07)="045" ' 45 MINUTES
0167	135C	PROVIDER.TIMES\$(08)="060" ' 1 HOUR
C170	135C	PROVIDER.TIMES\$(09)="999" ' >1 HOURS
0179	135C	
0179	135C	REM YES/NO TABLE
0179	135C	YNS\$(0)=" " : YNS\$(1)="Y" : YNS\$(2)="N" : YNS\$(3)="X"
019D	135C	
019D	135C	
019D	135C	REM *****
019D	135C	PNUM=VAL(PGMID\$)/10
01AF	135E	REM LENGTH OF STRING RECEIVED FROM THE OMR....
01AF	135E	HEADER = 21
0186	1360	RESPONSE= 309
019D	1362	RECORDLENGTH = HEADER + RESPONSE
01C3	1364	
01C8	1364	N.PROC = 25 ' NUMBER OF PROCEDURES FOR THIS FORM

RACP800

BAS/TMC FORM DESTING/DECODE PROGRAM

PAGE 5

05-26-87

09:33:12

IBM Personal Computer BAS/C Compiler V1.00

Offset	Data	Source Line
01CF	1366	N.DIAG.COL=4 ' NUMBER OF DX COLUMNS ON THIS FORM
01D6	1368	
01D6	1368	DATFIL\$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
01DF	136C	BTIME\$=TIME\$ 'SCAN START TIME
01E8	1370	
01E8	1370	REM *** ENCOUNTER FORM PROCEDURE TABLE ***
01E8	1370	F.NAMES="RACPROC." + PGMID\$
01F6	1374	OPEN F.NAMES\$ FOR INPUT AS #3
0207	1374	FOR I600=0 TO 30
020D	1374	INPUT #3,PROCD\$(I600)
0225	1376	IF PROCD\$(I600)="ZZZZZ" THEN GOTO 4
0241	1376	NEXT I600
0250	1376	4 CLOSE #3
0257	1376	
0257	1376	REM *** ENCOUNTER FORM DIAGNOSIS TABLE ***
0257	1376	F.NAMES="RACDIAG." + PGMID\$
0265	1376	OPEN F.NAMES\$ FOR INPUT AS #3
0276	1376	FOR I600 = 0 TO 160
027C	1376	INPUT #3,DIAGN.TAB\$(I600)
0294	1376	IF DIAGN.TAB\$(I600)="ZZZZZ" THEN GOTO 6
02B0	1376	NEXT I600
02C0	1376	6 CLOSE #3
02C7	1376	
02C7	1376	REM INCLUDE: 'UCACAMP.OPT' INCLUDE THE OUTP UCA VALIDATE TABLE
02C7	1376	REM INCLUDE: 'UCACAMP.IPT' INCLUDE THE INP UCA VALIDATE TABLE
02C7	1376	
02C7	1376	REM \$PAGE

Offset	Data	Source Line
02C7	1376	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
02CC	1376	CLS
02D0	1376	GOSUB 7000 'PRINT SCREEN HEADING
02D5	1376	
02D5	1376	REM *****
02D5	1376	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
02D5	1376	REM *****
02D5	1376	REM
02D5	1376	OPEN DATFIL\$ FOR APPEND AS #1
02E7	1376	
02E7	1376	REM *****
02E7	1376	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
02E7	1376	REM *****
02E7	1376	LPRINT CHR\$(15);
02F2	1376	WIDTH "LPT1:",160
02FC	1376	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
0308	1378	COLOR 14
030F	1378	LOCATE 11,26 : PRINT "BAS/TMC FORM "
0324	1378	COLOR FORE,BACK,BORD
033A	1378	
033A	1378	REM *****
033A	1378	REM **** COMMUNICATIONS SETUP ****
033A	1378	REM *****
033A	1378	REM PROTOCOL
033A	1378	GOSUB 9001
033F	1378	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0359	137C	
0359	137C	REM START SCANNER (S1)
0359	137C	CNTRLOPT = 1 : GOSUB 9010
0365	137E	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
037F	137E	
037F	137E	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
0394	137E	READTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
0398	1380	
0398	1380	REM *****
0398	1380	REM *** SET SCAN SHEET CALL ***
0398	1380	REM *****
0398	1380	REM
0398	1380	
039B	1380	10 REM - RETURN POINT TO READ NEXT SHEET
039C	1380	
039C	1380	AS=INKEY\$
03A5	1384	IF AS=CHR\$(27) THEN GOTO 25000
03BB	1384	
03BB	1384	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
03C0	1384	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
03DC	1388	
03DC	1388	TEXT\$="" 'CLEAR THE INPUT AREA
03E5	1388	GOSUB 8000 'DECODE HEADER
03EA	1388	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
03EF	1388	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
03F4	1388	LITHO\$ = MID\$(TEXT\$,22,8)
0406	138C	GOSUB 8070 'CHECK FOR SCANNER ERRORS
040B	138C	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

Offset	Data	Source Line
0410	138C	
0410	138C	REM \$INCLUDE: 'RAC800.MO1' INCLUDE THE BAS/TMC FORM REFORMAT/EDIT MOD
0410	138C	REM *****
0410	138C	REM **** AMBULATORY CARE INFORMATION SYSTEM 26 MAY 87 ****
0410	138C	REM **** D R BOLLING ****
0410	138C	REM **** MODULE NAME : RAC800.MO1 ****
0410	138C	REM **** SCANNER PROGRAM # : BAS/TMC (BHA) FORM ****
0410	138C	REM **** PURPOSE : REFORMAT/EDIT THE FORM ****
0410	138C	REM ****
0410	138C	REM *****
0410	138C	REM **** RESERVED LINE NUMBERS 100-199 ****
0410	138C	REM *****
0410	138C	
0410	138C	N.ERR =0 'COUNTS THE NUMBER OF ERRORS
0417	138E	
0417	138E	REM *** LITHO CODE DONE IN BAS PROGRAM ***
0417	138E	
0417	138E	REM *** CLINIC ID (PREFIX + CODE) ***
0417	138E	100 CL1.COD\$="BHA" 'DEFAULT CLINIC CODE
0420	1392	TCL1.COD\$="BHA"
0429	1396	
0429	1396	REM ... CLINIC CODE WHEN BAS BUBBLED ...
0429	1396	IF MID\$(TEXT\$,59,2)=" " THEN 102
0441	1396	CL1.COD\$="SB" +MID\$(TEXT\$,59,2)
0458	1396	
0458	1396	REM ... CLINIC CODE WHEN TMC CODED ...
0458	1396	102 IF MID\$(TEXT\$,88,2)=" " THEN 104
0470	1396	CK.X=VAL(MID\$(TEXT\$,87,1))
0486	1398	CK.COD\$=MID\$(TEXT\$,88,2)
0498	139C	TCL1.COD\$="B" + CLINIC1.PFX\$(CK.X) + CK.COD\$
0485	139C	IF TCL1.COD\$="BJAA" THEN TCL1.COD\$="BJYA"
04CC	139C	
04CC	139C	REM VISIT DATE
04CC	139C	104 CMS=MID\$(DATE\$,1,2) 'CURRENT MONTH
04DE	13A0	YRS=MID\$(DATE\$,9,2) 'CURRENT YEAR
04F0	13A4	XS=MID\$(TEXT\$,30,4) 'MONTH AND DAY
0502	13A8	IF LEFT\$(XS,2)<=CMS THEN 105 'OK, USE THIS YEAR
0517	13A8	YRS=RIGHT\$(STR\$(VAL(YRS)-1),2) 'USE LAST YEAR
0537	13A8	105 VIDATES=YRS+XS
0545	13AC	'EDIT VISIT DATE
0545	13AC	CK.5000\$=VIDATES
054E	13B0	GOSUB 5000 'DATE CHECK
0553	13B0	IF RT.5000=0 THEN GOTO 106
0562	13B2	N.ERR=N.ERR+1
056A	13B2	ED.MSG\$(N.ERR)="VISIT DATE" + DATEERR\$(RT.5000)
0590	13B2	
0590	13B2	REM *** PRIMARY PROVIDER ***
0590	13B2	106 PPROV.PFX\$=MID\$(TEXT\$,61,1) 'BAS
05A2	13B6	PPROV.NUM\$=MID\$(TEXT\$,62,4) 'BAS
05B4	13BA	TPPROV.PFX\$=MID\$(TEXT\$,90,1) 'TMC
05C6	13BE	TPPROV.NUM\$=MID\$(TEXT\$,91,4) 'TMC
05D8	13C2	IF PPROV.PFX\$+PPROV.NUM\$=" " THEN BAS=0 ELSE BAS=1
05FD	13C4	IF TPPROV.PFX\$+TPPROV.NUM\$=" " THEN TMC=0 ELSE TMC=1

Offset Data Source Line

```

0622 13C6
0622 13C6      IF TPROV.PFX$+TPROV.NUM$="      " AND BAS=0 THEN 107
064B 13C6      GOTO 108
064F 13C6
064F 13C6      107 N.ERR=N.ERR+1
0657 13C6      ED.MSG$(N.ERR)="NO PROVIDER CODED FOR BAS OR TMC"
066B 13C6
066B 13C6      REM *** PATIENT SSN ***
066B 13C6      108 SSN$ = MID$(TEXT$,34,9)
067D 13CA
067D 13CA      REM *** FAMILY MEMBER PREF ***
067D 13CA      110 FMEMP$="20"
0686 13CE
0686 13CE      REM *** PRIMARY PROVIDER TIME ***
0686 13CE      112 X=VAL(MID$(TEXT$,66,1)) 'BAS
069C 13D0      PPROV.TIMS=PROVIDER.TIMES(X) 'BAS
06AE 13D4      X=VAL(MID$(TEXT$,95,1)) 'TMC
06C4 13D4      TPROV.TIMS=PROVIDER.TIMES(X) 'TMC
06D6 13D8
06D6 13D8      REM *** SECONDARY PROVIDER ***
06D6 13D8      SPROV.PFX$=MID$(TEXT$,67,1) 'BAS
06E8 13D0      SPROV.NUM$=MID$(TEXT$,68,4) 'BAS
06FA 13E0      TSPROV.PFX$=MID$(TEXT$,96,1) 'TMC
070C 13E4      TSPROV.NUM$=MID$(TEXT$,97,4) 'TMC
071E 13E8      IF SPROV.PFX$+SPROV.NUM$="      " THEN PB=1 ELSE PB=0
0743 13EA      IF TSPROV.PFX$+TSPROV.NUM$="      " THEN TPB=1 ELSE TPB=0
0768 13EC      IF SPROV.PFX$="      " AND SPROV.NUM$<>"      " THEN 113
078E 13EC      IF SPROV.PFX$<>"      " AND SPROV.NUM$="      " THEN 113
07B4 13EC      IF TSPROV.PFX$="      " AND TSPROV.NUM$<>"      " THEN 114
07DA 13EC      IF TSPROV.PFX$<>"      " AND TSPROV.NUM$="      " THEN 114
0800 13EC      GOTO 115
0804 13EC
0804 13EC      113 N.ERR=N.ERR + 1
080C 13EC      ED.MSG$(N.ERR)="BAS PROV 2 CODE MISSING PREFIX OR NUMBER"
0820 13EC      GOTO 115
0824 13EC
0824 13EC      114 N.ERR=N.ERR + 1
082C 13EC      ED.MSG$(N.ERR)="TMC PROV 2 CODE MISSING PREFIX OR NUMBER"
0840 13EC
0840 13EC      REM *** SECONDARY PROVIDER TIME ***
0840 13EC      115 X=VAL(MID$(TEXT$,72,1)) 'BAS
0856 13EC      SPROV.TIMS=PROVIDER.TIMES(X) 'BAS
0868 13F0      X=VAL(MID$(TEXT$,101,1)) 'TMC
087E 13F0      TSPROV.TIMS=PROVIDER.TIMES(X) 'TMC
0890 13F4      REM IS THERE A TIME AND NO SEC PROV CODED?
0890 13F4      IF SPROV.TIMS<>"000" AND PB=1 THEN 116
08B3 13F4      REM IS THERE NO TIME AND A SEC PROV CODED?
08B3 13F4      IF SPROV.TIMS="000" AND PB=0 THEN 117
08D6 13F4      REM IS THERE A TIME AND NO SEC PROV CODED?
08D6 13F4      IF TSPROV.TIMS<>"000" AND TPB=1 THEN 118
08F9 13F4      REM IS THERE NO TIME AND A SEC PROV CODED?
08F9 13F4      IF TSPROV.TIMS="000" AND TPB=0 THEN 119
091C 13F4      GOTO 120
0920 13F4

```

Offset	Data	Source Line
0920	13F4	116 N.ERR=N.ERR + 1
0928	13F4	ED.MSG\$(N.ERR)="BAS TIME CODED WITH NO SEC PROV CODED"
093C	13F4	GOTO 120
0940	13F4	
0940	13F4	117 N.ERR=N.ERR + 1
0948	13F4	ED.MSG\$(N.ERR)="BAS NO PROV 2 TIME"
095C	13F4	GOTO 120
0960	13F4	
0960	13F4	118 N.ERR=N.ERR + 1
0968	13F4	ED.MSG\$(N.ERR)="TMC TIME CODED WITH NO SEC PROV CODED"
097C	13F4	GOTO 120
0980	13F4	
0980	13F4	119 N.ERR=N.ERR + 1
0988	13F4	ED.MSG\$(N.ERR)="TMC NO PROV 2 TIME"
099C	13F4	
099C	13F4	REM *** REASON FOR SECONDARY PROVIDER ***
099C	13F4	120 SPROV.REAS=MID\$(TEXT\$,73,1) 'BAS
09AE	13F8	TSPROV.REAS=MID\$(TEXT\$,102,1) 'TMC
09C0	13FC	
09C0	13FC	REM IS THERE A REAS AND NO SEC PROV CODED?
09C0	13FC	IF SPROV.REAS<>" " AND PB=1 THEN 121
09E3	13FC	REM IS THERE NO REAS AND A SEC PROV CODED?
09E3	13FC	IF SPROV.REAS="" " AND PB=0 THEN 122
0A06	13FC	REM IS THERE A REAS AND NO SEC PROV CODED?
0A06	13FC	IF TSPROV.REAS<>" " AND TPB=1 THEN 123
0A29	13FC	REM IS THERE NO REAS AND A SEC PROV CODED?
0A29	13FC	IF TSPROV.REAS="" " AND TPB=0 THEN 124
0A4C	13FC	GOTO 129
0A50	13FC	
0A50	13FC	121 N.ERR=N.ERR + 1
0A58	13FC	ED.MSG\$(N.ERR)="BAS REASON CODED WITH NO SEC PROV CODED"
0A6C	13FC	GOTO 129
0A70	13FC	
0A70	13FC	122 N.ERR=N.ERR + 1
0A78	13FC	ED.MSG\$(N.ERR)="BAS NO PROV 2 REASON"
0A8C	13FC	GOTO 129
0A90	13FC	
0A90	13FC	123 N.ERR=N.ERR + 1
0A98	13FC	ED.MSG\$(N.ERR)="TMC REASON CODED WITH NO SEC PROV CODED"
0AAC	13FC	GOTO 129
0AB0	13FC	
0AB0	13FC	124 N.ERR=N.ERR + 1
0AB8	13FC	ED.MSG\$(N.ERR)="TMC NO PROV 2 REASON"
0ACC	13FC	
0ACC	13FC	REM *** APPOINTMENT STATUS ***
0ACC	13FC	129 NOTSC\$="5" 'DEFAULT IS SICK CALL = 5
0AD5	1400	X\$=MID\$(TEXT\$,43,1)
0AE7	1400	IF X\$="1" THEN NOTSC\$="4" 'WALK IN
0AFE	1400	IF X\$="2" THEN NOTSC\$="1" 'SCHEDULED
0B15	1400	IF X\$="3" THEN NOTSC\$="3" 'EMERGENCY
0B2C	1400	
0B2C	1400	REM *** INJURIES ***
0B2C	1400	INJUR\$=MID\$(TEXT\$,44,1)
0B3E	1404	

Offset	Data	Source Line
083E	1404	REM *** JOB RELATED VISIT ***
083E	1404	RELAT.VISS="N"
0847	1408	IF MIDS(TEXT\$,45,1)="1" THEN RELAT.VISS="Y"
0868	1408	
0868	1408	REM *** MIL ONLY DUTY ***
0868	1408	REM *** DO SEPARATELY FOR BAS ***
0868	1408	MILQTRS=" "
0871	140C	MILPROS=" "
087A	1410	MILDUTS=" "
0883	1414	B.QTRS=" " 'NO QTR FOR BAS
088C	1418	B.PROS=" " 'NO PROF FOR BAS
0895	141C	IF BAS = 0 THEN 133 'GO IF BAS SECTION NOT FILLED IN
08A0	141C	IF TMC = 0 THEN 130 'GO IF TMC SECTION NOT FILLED IN
08AB	141C	B.DUTS="3" 'MAKE TMC BY DEFAULT
08B4	1420	GOTO 133
08B8	1420	
08B8	1420	130 XS=MIDS(TEXT\$,49,1) 'GET TMC BUBBLE
08CA	1420	IF XS=" " THEN GOTO 131
08DC	1420	B.DUTS="3" 'MAKE TMC BY DEFAULT
08E5	1420	XS=MIDS(TEXT\$,46,1) 'GET OTHER DUTY BUBBLES
08F7	1420	IF XS=" " THEN GOTO 140 'EVERYTHING OK, GO ON
0C09	1420	
0C09	1420	N.ERR=N.ERR + 1
0C11	1420	ED.MSG\$(N.ERR)="TMC CONFLICT IN DISPOSITION BOX"
0C25	1420	GOTO 133
0C29	1420	
0C29	1420	131 REM NO TMC BUBBLE CODED
0C2A	1420	XS=MIDS(TEXT\$,46,1) 'GET OTHER DUTY BUBBLES
0C3C	1420	IF XS=" " THEN B.DUTS="1" :GOTO 140 'MAKE DUTY DEFAULT
0C57	1420	IF XS="3" THEN B.DUTS="4" ELSE B.DUTS=XS 'GIVE PROPER CODE
0C7A	1420	B.QTRS=MIDS(TEXT\$,47,1)
0C8C	1420	B.PROS=MIDS(TEXT\$,48,1)
0C9E	1420	GOTO 140
0CA2	1420	
0CA2	1420	REM *** MIL ONLY DUTY ***
0CA2	1420	REM *** DO FOR TMC NOW ***
0CA2	1420	133 XS=MIDS(TEXT\$,49,1) 'GET TMC BUBBLE
0CB4	1420	IF XS=" " THEN GOTO 134
0CC6	1420	
0CC6	1420	IF BAS=1 THEN 134
0CD1	1420	N.ERR=N.ERR + 1
0CD9	1420	ED.MSG\$(N.ERR)="NO TMC CAN BE CODED IF NO BAS SECTION CODED"
0CED	1420	
0CED	1420	134 XS=MIDS(TEXT\$,46,1) 'GET DUTY BUBBLES
0CFF	1420	XAS=MIDS(TEXT\$,109,1) 'GET ADMITTED BUBBLE
0D11	1424	IF XS=" " THEN 135 'GO IF NO DUTY CODED
0D1F	1424	IF XS="3" THEN MILDUTS="4" ELSE MILDUTS=XS 'GIVE PROPER CODE
0D42	1424	GOTO 136
0D46	1424	135 IF XAS="1" THEN MILDUTS=" " :GOTO 140 'DONT CODE IF ADMITTED
0D61	1424	MILDUTS="1" 'GIVE DUTY BY DEFAULT
0D6A	1424	
0D6A	1424	REM *** MIL ONLY QTRS ***
0D6A	1424	136 MILDUTS=MIDS(TEXT\$,47,1)

RACP800

BAS/TMC FORM DESTURING/DECODE PROGRAM

PAGE 11

05-26-87

09:33:12

IBM Personal Computer BASIC Compiler v1.00

Offset Data Source Line

```

007C 1424 REM *** MIL ONLY PROF ***
007C 1424 MILPROS=MID$(TEXT$,48,1)
008E 1424
008E 1424 REM *** NOT AVAILABLE ***
008E 1424 140 NAVAILS=" "
0097 1428
0097 1428 REM *** SPEC PREASSIGNED CLINIC ***
0097 1428 INP.STOS=MID$(TEXT$,50,9)
00A9 142C INP.STOS=INP.STOS + MID$(TEXT$,328,3) 'ADD NOT AVAIL ARRAY
00BE 142C GOSUB 5700 'CONVERT ARRAY
00C3 142C SPE.BUF$=BUF.STOS 'UP TO 9 TWO DIGIT CODES
00CC 1434
00CC 1434 REM *** BAS PROVIDER 2 ADDL PROC 1 ***
00CC 1434 ADDP1$=" " : ADDP2$=" " 'INITIALIZE
00DE 143C IF BAS=0 THEN 148 'NO NEED TO DO IF BAS BLANK
00E9 143C 144 PR2PRC1$="N"
00F2 1440 IF MID$(TEXT$,74,1)="1" THEN PR2PRC1$="Y"
0E13 1440
0E13 1440 REM *** BAS ADDITIONAL PROCEDURE 1 ***
0E13 1440 ADDP1$=MID$(TEXT$,75,5)
0E25 1440 IF PRC2PRC1$="Y" AND PB=1 THEN 145
0E48 1444
0E48 1444 REM *** BAS PROVIDER 2 ADDL PROC 2 ***
0E48 1444 PR2PRC2$="N"
0E51 1448 IF MID$(TEXT$,80,1)="1" THEN PR2PRC2$="Y"
0E72 1448
0E72 1448 REM *** BAS ADDITIONAL PROCEDURE 2 ****
0E72 1448 ADDP2$=MID$(TEXT$,81,5)
0E84 1448 IF PRC2PRC2$="Y" AND PB=1 THEN 145
0EA7 144C GOTO 148
0EAB 144C
0EAB 144C 145 N.ERR=N.ERR + 1
0EB3 144C ED.MSG$(N.ERR)="BAS ADDL PROC CODED FOR PROV 2 BUT PROV2 NOT CODED"
0EC7 144C
0EC7 144C REM *** TMC PROVIDER 2 ADDL PROC 1 ***
0EC7 144C 148 TPR2PRC1$="N"
0ED0 1450 IF MID$(TEXT$,103,1)="1" THEN TPR2PRC1$="Y"
0EF1 1450
0EF1 1450 REM *** TMC ADDITIONAL PROCEDURE 1 ***
0EF1 1450 TADDP1$=MID$(TEXT$,104,5)
0F03 1454 IF TPRC2PRC1$="Y" AND PB=1 THEN 149
0F26 1458 GOTO 150
0F2A 1458
0F2A 1458 149 N.ERR=N.ERR + 1
0F32 1458 ED.MSG$(N.ERR)="TMC ADDL PROC CODED FOR PROV 2 BUT PROV2 NOT CODED"
0F46 1458
0F46 1458 REM *** ADMITTED ***
0F46 1458 150 ADMIT$="N"
0F4F 145C IF MID$(TEXT$,109,1)="1" THEN ADMIT$="Y"
0F70 145C
0F70 145C REM *** EDIT FOR MIL ONLY BOX ***
0F70 145C IF ADMIT$="N" THEN 152
0F7E 145C IF MILDUT$=" " AND MILOTR$=" " THEN 152
0FA1 145C

```

Offset	Data	Source Line
0FA1	145C	151 REM ... ERROR DUTY, QTRS AND ADMITTED CANNOT ALL BE CODED
0FA2	145C	N.ERR = N.ERR + 1
0FAA	145C	ED.MSG\$(N.ERR)="DUTY OR QTRS CANNOT BE CODED WITH ADMITTED"
0FBE	145C	
0FBE	145C	REM *** FIELD ***
0FBE	145C	152 FIELD\$="H"
0FC7	1460	IF MID\$(TEXT\$,86,1)="1" THEN FIELD\$="Y"
0FEB	1460	
0FEB	1460	REM *** UNLISTED PRIMARY DX ***
0FE8	1460	154 X=VAL(MID\$(TEXT\$,161,1))
0FFE	1460	IF X=0 THEN PRIMDX\$=""
1012	1464	IF X=1 THEN PRIMDX\$="V"
1026	1464	IF X=2 THEN PRIMDX\$="S"
103A	1464	DXTMP\$=MID\$(TEXT\$,162,5)
104C	1468	IF DXTMP\$="" THEN GOTO 158
105E	1468	
105E	1468	REM REMOVE LEADING BLANKS
105E	1468	156 IF LEFT\$(DXTMP\$,1)="" THEN DXTMP\$=RIGHT\$(DXTMP\$,4)+" ":GOTO 156
108D	1468	
108D	1468	158 LASTC\$=RIGHT\$(DXTMP\$,1) 'GET LAST CHAR
109C	146C	IF LASTC\$<>" " AND LASTC\$<>"0" THEN PRIMDX\$="" 'REMOVE V OR S
10C8	146C	PRIMDX\$=LEFT\$(PRIMDX\$+DXTMP\$,5) 'GET 5 CHAR ONLY
10DA	146C	
10DA	146C	REM *** UNLISTED SECONDARY DX ***
10DA	146C	160 X=VAL(MID\$(TEXT\$,167,1))
10F0	146C	IF X=0 THEN SECDX\$=""
1104	1470	IF X=1 THEN SECDX\$="V"
1118	1470	IF X=2 THEN SECDX\$="S"
112C	1470	DXTMP\$=MID\$(TEXT\$,168,5)
113E	1470	IF DXTMP\$="" THEN GOTO 164
1150	1470	
1150	1470	REM REMOVE LEADING BLANKS
1150	1470	162 IF LEFT\$(DXTMP\$,1)="" THEN DXTMP\$=RIGHT\$(DXTMP\$,4)+" ":GOTO 162
117F	1470	
117F	1470	164 LASTC\$=RIGHT\$(DXTMP\$,1) 'GET LAST CHAR
118E	1470	IF LASTC\$<>" " AND LASTC\$<>"0" THEN SECDX\$="" 'REMOVE V OR S
118A	1470	SECDX\$=LEFT\$(SECDX\$+DXTMP\$,5) 'GET 5 CHAR ONLY
11CC	1470	
11CC	1470	REM *** FOLLOW UP/RULE OUT ***
11CC	1470	166 FU.ROS=MID\$(TEXT\$,160,1)
11DE	1474	
11DE	1474	REM *** EVALUATION/SERV/PROC PROV 1 ***
11DE	1474	REM *** TMC ONLY ***
11DE	1474	168 ESP.BUF1\$=""
11E7	1478	ESP.BUF2\$=""
11F0	147C	INP.STO\$=MID\$(TEXT\$,110,N.PROC)
1203	147C	N.STO=1
120A	147E	
120A	147E	170 X.STO=INSTR(N.STO,INP.STO\$,"1")
121C	1480	IF X.STO=0 THEN GOTO 172 'THATS ALL
122B	1480	N.STO=X.STO + 1 'NEXT STARTING POINT
1233	1480	ESP.BUF1\$=ESP.BUF1\$ + PROCED\$(X.STO) 'ADD CODE TO BUFFER BY FIVES
1240	1480	IF N.STO <= LEN(INP.STO\$) THEN GOTO 170
125A	1480	

Offset Data Source Line

```

1258 1480 REM *** EVALUATION/SERV/PROC PROV 2 ***
1258 1480 172 INP.STOS=MID$(TEXT$,110+N.PROC,N.PROC)
1272 1480 N.STO=1
1279 1480
1279 1480 174 X.STO=INSTR(N.STO,INP.STOS,"1")
1288 1480 IF X.STO=0 THEN GOTO 176 'THATS ALL
129A 1480 IF TPB=1 THEN 175
12A5 1480 N.STO=X.STO + 1 'NEXT STARTING POINT
12AD 1480 ESP.BUF2$=ESP.BUF2$ + PROCED$(X.STO) 'ADD CODE TO BUFFER BY FIVES
12C2 1480 IF N.STO <= LEN(INP.STOS) THEN GOTO 174
12D5 1480 GOTO 176
12D9 1480
12D9 1480 175 N.ERR = N.ERR + 1
12E1 1480 ED.MSG$(N.ERR)="PROC CODED FOR PROV 2 BUT PROV 2 NOT CODED"
12F5 1480
12F5 1480 176 T.POS = 173
12FC 1482
12FC 1482 184 REM
12FD 1482
12FD 1482 REM *** PRIMARY DX ***
12FD 1482 REM * IF OTHER PRIM DX IS CODED THEN SKIP THIS SECTION *
12FD 1482 186 IF PRIMDX$<>" " THEN GOTO 192
130F 1482 X.POS=T.POS 'STARTING POSITION
1316 1484 C1SIZ=37 : C3SIZ=42 'NO. OF ITEMS IN EACH COL
1324 1488 C2SIZ=30 : C4SIZ=40
1332 148C GOSUB 5800 'GET POSITION
1337 148C IF X.FIN=0 THEN GOTO 188
1346 148E IF RT.5800=0 THEN GOTO 190
1355 1490 N.ERR=N.ERR + 1
135D 1490 ED.MSG$(N.ERR)="PRIMARY DX HAS MULTIPLE CODES"
1371 1490 GOTO 190
1375 1490
1375 1490 188 N.ERR=N.ERR + 1
137D 1490 ED.MSG$(N.ERR)="PRIMARY DX NOT CODED"
1391 1490
1391 1490 190 PRIMDX$=DIAGN.TAB$(X.FIN)
13A3 1490
13A3 1490 192 T.POS = T.POS + 2 * N.DIAG.COL
1382 1490 NDX = 147
1389 1492
1389 1492 REM *** SECONDARY DX ***
1389 1492 194 INP.STOS=MID$(TEXT$,T.POS,NDX)
13CD 1492 DX2.BUF$=""
1306 1496 N.STO=1
130D 1496
130D 1496 196 X.STO=INSTR(N.STO,INP.STOS,"1")
13EF 1496 IF X.STO=0 THEN GOTO 197 'THATS ALL
13FE 1496 N.STO=X.STO + 1 'NEXT STARTING POINT
1406 1496 DX2.BUF$=DX2.BUF$ + DIAGN.TAB$(X.STO) 'ADD CODE TO BUFFER BY FIVES
1418 1496 IF N.STO <= LEN(INP.STOS) THEN GOTO 196
142E 1496 197 IF SECDX$=" " THEN GOTO 198
1440 1496 DX2.BUF$=DX2.BUF$+SECDX$ 'ADD OTHER SEC DX IF THERE
144C 1496
144C 1496 198 REM

```

```

Offset  Data  Source Line
144D 1496 REM -----END OF MODULE RACM800.MO1-----
144D 1496
144D 1496 IF N.ERR = 0 THEN GOTO 997
145C 1496 LPRINT "LITHO # ";LITHOS;" ...ERRORS"
146E 1496 FOR I997 = 1 TO N.ERR
147B 1498 LPRINT USING "### ";I997;
1487 149A LPRINT "==> ";ED.MSG$(I997)
149D 149A NEXT I997
14AE 149A LN.COUNT = LN.COUNT + N.ERR + 1
148A 149C CNTRLOPT = 6
14C1 149C GOSUB 9010 'REJECT THE FORM
14C6 149C GOTO 998 'BYPASS THE DISK WRITER....
14CA 149C
14CA 149C 997 REM $INCLUDE: 'RACM800.MO2' REM INCLUDE THE BASE FORM DISK WRITER
14CB 149C REM *****
14CB 149C REM **** AMBULATORY CARE INFORMATION SYSTEM 16 APR 87 ****
14CB 149C REM **** D R BOLLING ****
14CB 149C REM **** MODULE NAME : RACM800.MO2 ****
14CB 149C REM **** SCANNER PROGRAM # : BAS/TMC FORM ****
14CB 149C REM ****
14CB 149C REM **** PURPOSE : CREATE AND WRITE THE DISK ****
14CB 149C REM **** RECORD FOR INPUT TO FOCUS ****
14CB 149C REM ****
14CB 149C REM *****
14CB 149C REM **** RESERVED LINE NUMBERS 200-299 ****
14CB 149C REM *****
14CB 149C
14CB 149C REM BUILD THE OUTPUT RECORD
14CB 149C
14CB 149C GOSUB 270 'BUILD THE RECORD KEY FOR BAS
14D0 149C GOSUB 271 'BUILD THE RECORD KEY FOR TMC
14D5 149C
14D5 149C REM *****
14D5 149C REM **** RECORD TYPE "1" - MAIN TRANSACTION ****
14D5 149C REM *****
14D5 149C REM BAS FIRST
14D5 149C
14D5 149C IF PRV$=" " THEN 200 'DONT DO IF NO PROV
14E3 14A0 GOSUB 272 'BUILD DATA FOR TYPE 1 BAS
14E8 14A0 RECOU$ =PGMID$+"1"+RECKEY$+RECOD1$ 'TRANSACTION ID PLUS RECORD
1502 14AC GOSUB 280
1507 14AC PRINT #1,RECOU$
1512 14AC
1512 14AC 200 REM NOW DO TMC
1513 14AC
1513 14AC IF TPRV$=" " THEN 201 'DONT DO IF NO PROV
1521 14B0 GOSUB 273 'BUILD DATA FOR TYPE 1 TMC
1526 14B0 RECOU$ =PGMID$+"1"+TRECKEY$+RECOD1$ 'TRANSACTION ID PLUS RECORD
1540 14B4 GOSUB 280
1545 14B4 PRINT #1,RECOU$
1550 14B4
1550 14B4 REM *****
1550 14B4 REM **** RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE ****
1550 14B4 REM *****

```

Offset	Data	Source Line
1550	14B4	201 REM BEGIN WITH BAS
1551	14B4	IF PRV\$=" " THEN 212 'DONT DO IF NO PROV
155F	14B4	BUF1\$="" : BUF2\$="" 'INITIALIZE
1571	14BC	
1571	14BC	REM *** GET ADDITIONAL PROCEDURES IF ANY ***
1571	14BC	IF ADDP1\$<>" " AND PR2PRC1\$="N" THEN BUF1\$=BUF1\$+ADDP1\$
15A3	14BC	IF ADDP2\$<>" " AND PR2PRC2\$="N" THEN BUF1\$=BUF1\$+ADDP2\$
15D5	14BC	
15D5	14BC	IF ADDP1\$<>" " AND PR2PRC1\$="Y" THEN BUF2\$=BUF2\$+ADDP1\$
1607	14BC	IF ADDP2\$<>" " AND PR2PRC2\$="Y" THEN BUF2\$=BUF2\$+ADDP2\$
1639	14BC	
1639	14BC	REM *** PROCESS PROV 1 PROCEDURES ***
1639	14BC	
1639	14BC	IF LEN(BUF1\$)=0 THEN GOTO 206
164A	14BC	
164A	14BC	RPOINT=1
1651	14BE	202 RECOD2\$=MID\$(BUF1\$,RPOINT,5)
1664	14C2	IF RECOD2\$=" " THEN GOTO 204
1676	14C2	
1676	14C2	RECOU\$=PGMID\$+"2"+RECKEY\$+"1"+RECOD2\$ 'TRANSACTION ID PLUS RECORD
1696	14C2	GOSUB 280
1698	14C2	PRINT #1,RECOU\$
16A6	14C2	
16A6	14C2	204 RPOINT = RPOINT + 5
1680	14C2	IF RPOINT < LEN(BUF1\$) THEN GOTO 202
16C3	14C2	
16C3	14C2	REM *** PROCESS PROV 2 PROCEDURES ***
16C3	14C2	
16C3	14C2	206 IF LEN(BUF2\$)=0 THEN GOTO 212
16D4	14C2	
16D4	14C2	RPOINT=1
16D8	14C2	208 RECOD2\$=MID\$(BUF2\$,RPOINT,5)
16EE	14C2	IF RECOD2\$=" " THEN GOTO 210
1700	14C2	
1700	14C2	RECOU\$=PGMID\$+"2"+RECKEY\$+"2"+RECOD2\$ 'TRANSACTION ID PLUS RECORD
1723	14C2	GOSUB 280
1728	14C2	PRINT #1,RECOU\$
1733	14C2	
1733	14C2	210 RPOINT = RPOINT + 5
173D	14C2	IF RPOINT < LEN(BUF2\$) THEN GOTO 208
1750	14C2	
1750	14C2	REM NOW DO PROCEDURES FOR TMC *****
1750	14C2	212 IF TPRV\$=" " THEN 224 'DONT DO IF NO PROV
175E	14C2	
175E	14C2	REM *** ADD ADDITIONAL PROCEDURES IF ANY ***
175E	14C2	IF TADDP1\$<>" " AND TPR2PRC1\$="N" THEN ESP.BUF1\$=ESP.BUF1\$+TADDP1\$
1790	14C2	
1790	14C2	IF TADDP1\$<>" " AND TPR2PRC1\$="Y" THEN ESP.BUF2\$=ESP.BUF2\$+TADDP1\$
17C2	14C2	
17C2	14C2	REM *** PROCESS PROV 1 PROCEDURES TMC ***
17C2	14C2	
17C2	14C2	IF LEN(ESP.BUF1\$)=0 THEN GOTO 218
17D3	14C2	
17D3	14C2	RPOINT=1

Offset	Data	Source Line
17DA	14C2	214 RECOD2\$=MID\$(ESP.BUF1\$,RPOINT,5)
17ED	14C2	IF RECOD2\$="" THEN GOTO 216
17FF	14C2	
17FF	14C2	RECOUT\$=PGMID\$+"2"+TRECKEY\$+"1"+RECOD2\$ 'TRANSACTION ID PLUS RECORD
181F	14C2	GOSUB 280
1824	14C2	PRINT #1,RECOUT\$
182F	14C2	
182F	14C2	216 RPOINT = RPOINT + 5
1839	14C2	IF RPOINT < LEN(ESP.BUF1\$) THEN GOTO 214
184C	14C2	
184C	14C2	REM *** PROCESS PROV 2 PROCEDURES TMC ***
184C	14C2	
184C	14C2	218 IF LEN(ESP.BUF2\$)=0 THEN GOTO 224
185D	14C2	
185D	14C2	RPOINT=1
1864	14C2	220 RECOD2\$=MID\$(ESP.BUF2\$,RPOINT,5)
1877	14C2	IF RECOD2\$="" THEN GOTO 222
1889	14C2	
1889	14C2	RECOUT\$=PGMID\$+"2"+TRECKEY\$+"2"+RECOD2\$ 'TRANSACTION ID PLUS RECORD
18AC	14C2	GOSUB 280
18B1	14C2	PRINT #1,RECOUT\$
188C	14C2	
188C	14C2	222 RPOINT = RPOINT + 5
18C6	14C2	IF RPOINT < LEN(ESP.BUF2\$) THEN GOTO 220
18D9	14C2	
18D9	14C2	224 REM END OF TYPE 2 RECORDS
18DA	14C2	
18DA	14C2	REM *****
18DA	14C2	REM **** RECORD TYPE "3" - RECKEY PLUS SPECIFIC PRE CLINIC CODES ****
18DA	14C2	REM *****
18DA	14C2	REM DONT DO FOR BAS
18DA	14C2	
18DA	14C2	254 REM *** DO FOR TMC ***
18DB	14C2	IF TPRV\$="" THEN 260 'DONT DO IF NO PROV
18E9	14C2	IF LEN(SPE.BUF\$)=0 THEN 260
18F6	14C2	
18F6	14C2	RPOINT=1
18FD	14C2	256 X3\$=MID\$(SPE.BUF\$,RPOINT,2)
1910	14C6	IF X3\$="10" THEN RECOD3\$="M" :GOTO 257
1928	14CA	IF X3\$="11" THEN RECOD3\$="L" :GOTO 257
1946	14CA	IF X3\$="12" THEN RECOD3\$="X" :GOTO 257
1961	14CA	
1961	14CA	RECOD3\$=RIGHT\$(X3\$,1)
1970	14CA	IF RECOD3\$="" THEN 258
197E	14CA	
197E	14CA	
197E	14CA	257 RECOUT\$=PGMID\$ + "3" + TRECKEY\$ + RECOD3\$
1978	14CA	GOSUB 280
1990	14CA	PRINT #1,RECOUT\$
19A8	14CA	
19A8	14CA	258 RPOINT=RPOINT+2
19B1	14CA	IF RPOINT < LEN(SPE.BUF\$) THEN 256
19C0	14CA	
19C0	14CA	260 REM END OF TYPE 3 RECORDS

Offset	Data	Source Line
19C1	14CA	
19C1	14CA	REM *****
19C1	14CA	REM **** RECORD TYPE "7" - RECKEY PLUS SECOND DX CODE ****
19C1	14CA	REM *****
19C1	14CA	
19C1	14CA	IF LEN(DX2.BUF\$)=0 THEN GOTO 268
19D2	14CA	
19D2	14CA	RPOINT=1
19D9	14CA	264 RECOD7\$=MID\$(DX2.BUF\$,RPOINT,5)
19EC	14CE	IF RECOD7\$="" THEN GOTO 266
19FE	14CE	
19FE	14CE	REM *** FOR BAS SECTION ***
19FE	14CE	IF PRV\$="" THEN 265 'DONT DO IF NO PROV
1A0C	14CE	RECOU7\$=PGMID\$+"7"+RECKEY\$+RECOD7\$ 'TRANSACTION ID 7
1A26	14CE	GOSUB 280
1A28	14CE	PRINT #1,RECOU7\$
1A36	14CE	
1A36	14CE	REM *** FOR TMC SECTION ***
1A36	14CE	265 IF TPRV\$="" THEN 266 'DONT DO IF NO PROV
1A44	14CE	RECOU7\$=PGMID\$+"7"+TRECKEY\$+RECOD7\$ 'TRANSACTION ID 7
1A5E	14CE	GOSUB 280
1A63	14CE	PRINT #1,RECOU7\$
1A6E	14CE	
1A6E	14CE	266 RPOINT = RPOINT + 5
1A78	14CE	IF RPOINT < LEN(DX2.BUF\$) THEN GOTO 264
1A88	14CE	
1A88	14CE	268 REM END OF TYPE 7 RECORDS
1A8C	14CE	
1A8C	14CE	GOTO 299
1A90	14CE	
1A90	14CE	REM *****
1A90	14CE	REM **** SUBROUTINE 270 - BUILD THE RECORD KEY FOR BAS ****
1A90	14CE	REM *****
1A90	14CE	270 RECKEY\$=""
1A99	14CE	
1A99	14CE	PRV\$ = PPROV.PFX\$ + PPROV.NUM\$
1AA7	14CE	IF PRV\$ = "" THEN RETURN 'DONT DO IF NOT CODED
1AB8	14CE	
1AB8	14CE	REM *** CLINIC ID (PREFIX + COD) ***
1AB8	14CE	RECKEY\$= CL1.COD\$
1AC1	14CE	
1AC1	14CE	REM *** VISIT DATE ***
1AC1	14CE	RECKEY\$=RECKEY\$+ VDATE\$
1ACD	14CE	
1ACD	14CE	REM *** PRIMARY PROVIDER ***
1ACD	14CE	RECKEY\$ = RECKEY\$ + PPROV.PFX\$ + PPROV.NUM\$
1AE0	14CE	
1AE0	14CE	REM *** PATIENT SSN ***
1AE0	14CE	RECKEY\$ = RECKEY\$ + SSN\$
1AEC	14CE	
1AEC	14CE	REM *** FAMILY MEMBER PREF ***
1AEC	14CE	RECKEY\$ = RECKEY\$ + FMEMP\$
1AF8	14CE	
1AF8	14CE	REM *** LITHO CODE ***

Offset	Data	Source Line
1AF8	14CE	RECKEY\$ = RECKEY\$ + LITHOS
1B04	14CE	
1B04	14CE	REM *** FORM NUMBER ***
1B04	14CE	RECKEY\$ = RECKEY\$ + "80"
1B10	14CE	
1B10	14CE	RETURN
1B13	14CE	
1B13	14CE	REM *****
1B13	14CE	REM **** SUBROUTINE 271 - BUILD THE RECORD KEY FOR TMC ****
1B13	14CE	REM *****
1B13	14CE	271 TRECKEY\$=""
1B1C	14CE	
1B1C	14CE	TPRV\$ = TPPROV.PFX\$ + TPPROV.NUM\$
1B2A	14CE	IF TPRV\$ = " " THEN RETURN 'DONT DO IF NOT CODED
1B3B	14CE	
1B3B	14CE	REM *** CLINIC ID (PREFIX + COD) ***
1B3B	14CE	TRECKEY\$= TCL1.COD\$
1B44	14CE	
1B44	14CE	REM *** VISIT DATE ***
1B44	14CE	TRECKEY\$=TRECKEY\$+ VIDATES
1B50	14CE	
1B50	14CE	REM *** PRIMARY PROVIDER ***
1B50	14CE	TRECKEY\$ = TRECKEY\$ + TPRV\$
1B5C	14CE	
1B5C	14CE	REM *** PATIENT SSN ***
1B5C	14CE	TRECKEY\$ = TRECKEY\$ + SSN\$
1B68	14CE	
1B68	14CE	REM *** FAMILY MEMBER PREF ***
1B68	14CE	TRECKEY\$ = TRECKEY\$ + FMEMP\$
1B74	14CE	
1B74	14CE	REM *** LITHO CODE ***
1B74	14CE	TRECKEY\$ = TRECKEY\$ + LITHOS
1B80	14CE	
1B80	14CE	REM *** FORM NUMBER ***
1B80	14CE	TRECKEY\$ = TRECKEY\$ + "80"
1B8C	14CE	
1B8C	14CE	RETURN
1B8F	14CE	
1B8F	14CE	REM *****
1B8F	14CE	REM **** SUBROUTINE 272 - BUILD THE DATA FOR TYPE 1 BAS ****
1B8F	14CE	REM *****
1B8F	14CE	272 RECOD1\$=""
1B98	14CE	
1B98	14CE	REM *** VISIT COUNT ***
1B98	14CE	RECOD1\$ = RECOD1\$ + "1" 'DEFAULT COUNT IS 1
19A4	14CE	
19A4	14CE	REM *** PRIMARY PROV TIME ***
19A4	14CE	RECOD1\$ = RECOD1\$ + PPROV.TIME\$
1B80	14CE	
1B80	14CE	REM *** SECONDARY PROVIDER ***
1B80	14CE	RECOD1\$ = RECOD1\$ + SPROV.PFX\$ + SPROV.NUM\$
1B83	14CE	
1B83	14CE	REM *** SECONDARY PROVIDER TIME ***
1B83	14CE	RECOD1\$ = RECOD1\$ + SPROV.TIME\$

Offset	Data	Source Line
18CF	14CE	
18CF	14CE	REM *** REASON FOR SECONDARY PROVIDER ***
18CF	14CE	RECOD1\$ = RECOD1\$ + SPROV.REAS
1808	14CE	
1808	14CE	REM *** APPOINTMENT STATUS ***
1808	14CE	RECOD1\$ = RECOD1\$ + "5" 'ALWAY SICK CALL FOR BAS
18E7	14CE	
18E7	14CE	REM *** REFERRAL CODE ***
18E7	14CE	RECOD1\$ = RECOD1\$ + " " 'NO REF CODE FOR THIS FORM
18F3	14CE	
18F3	14CE	REM *** PLACE OF VISIT ***
18F3	14CE	RECOD1\$ = RECOD1\$ + " " 'NO PLACE OF VISIT FOR THIS FORM
18FF	14CE	
18FF	14CE	REM *** JOB RELATED VISIT ***
18FF	14CE	RECOD1\$ = RECOD1\$ + RELAT.VISS
1C08	14CE	
1C08	14CE	REM *** MIL ONLY DUTY ***
1C08	14CE	RECOD1\$ = RECOD1\$ + B.DUT\$
1C17	14CE	
1C17	14CE	REM *** MIL ONLY QTRS ***
1C17	14CE	RECOD1\$ = RECOD1\$ + B.QTRS
1C23	14CE	
1C23	14CE	REM *** MIL ONLY PROFILE ***
1C23	14CE	RECOD1\$ = RECOD1\$ + B.PROS
1C2F	14CE	
1C2F	14CE	REM *** NOT AVAILABLE ***
1C2F	14CE	RECOD1\$ = RECOD1\$ + " " 'BLANK FOR BAS
1C38	14CE	
1C38	14CE	REM *** ADMITTED ***
1C38	14CE	RECOD1\$ = RECOD1\$ + "N" 'NO ADMITTED ON BAS
1C47	14CE	
1C47	14CE	REM *** INFIELD ***
1C47	14CE	RECOD1\$ = RECOD1\$ + FIELD\$
1C53	14CE	
1C53	14CE	REM *** INJURY ***
1C53	14CE	RECOD1\$ = RECOD1\$ + INJUR\$
1C5F	14CE	
1C5F	14CE	REM *** PURPOSE OF VISIT ***
1C5F	14CE	RECOD1\$ = RECOD1\$ + " "
1C68	14CE	
1C68	14CE	REM *** PRIM FOLLOW-UP/RULE OUT ***
1C68	14CE	RECOD1\$ = RECOD1\$ + FU.ROS
1C77	14CE	
1C77	14CE	REM *** PRIMARY DX CODE ***
1C77	14CE	RECOD1\$ = RECOD1\$ + PRIMDX\$
1C83	14CE	
1C83	14CE	RETURN
1C86	14CE	
1C86	14CE	REM *****
1C86	14CE	REM **** SUBROUTINE 273 - BUILD THE DATA FOR TYPE 1 TMC ****
1C86	14CE	REM *****
1C86	14CE	
1C86	14CE	273 RECOD1\$=""
1C8F	14CE	

Offset	Data	Source Line
1C8F	14CE	REM *** VISIT COUNT ***
1C8F	14CE	RECOD1\$ = RECOD1\$ + "1" 'DEFAULT COUNT IS 1
1C98	14CE	
1C98	14CE	REM *** PRIMARY PROV TIME ***
1C98	14CE	RECOD1\$ = RECOD1\$ + TSPROV.TIM\$
1CA7	14CE	
1CA7	14CE	REM *** SECONDARY PROVIDER ***
1CA7	14CE	RECOD1\$ = RECOD1\$ + TSPROV.PFX\$ + TSPROV.NUM\$
1C8A	14CE	
1C8A	14CE	REM *** SECONDARY PROVIDER TIME ***
1C8A	14CE	RECOD1\$ = RECOD1\$ + TSPROV.TIM\$
1CC6	14CE	
1CC6	14CE	REM *** REASON FOR SECONDARY PROVIDER ***
1CC6	14CE	RECOD1\$ = RECOD1\$ + TSPROV.REAS
1CD2	14CE	
1CD2	14CE	REM *** APPOINTMENT STATUS ***
1CD2	14CE	RECOD1\$ = RECOD1\$ + NOTSC\$
1CDE	14CE	
1CDE	14CE	REM *** REFERRAL CODE ***
1CDE	14CE	RECOD1\$ = RECOD1\$ + " " 'NO REF CODE FOR THIS FORM
1CEA	14CE	
1CEA	14CE	REM *** PLACE OF VISIT ***
1CEA	14CE	RECOD1\$ = RECOD1\$ + " " 'NO PLACE OF VISIT FOR THIS FORM
1CF6	14CE	
1CF6	14CE	REM *** JOB RELATED VISIT ***
1CF6	14CE	RECOD1\$ = RECOD1\$ + RELAT.VISS
1D02	14CE	
1D02	14CE	REM *** MIL ONLY DUTY ***
1D02	14CE	RECOD1\$ = RECOD1\$ + MILDUTS
1D0E	14CE	
1D0E	14CE	REM *** MIL ONLY QTRS ***
1D0E	14CE	RECOD1\$ = RECOD1\$ + MILQTRS
1D1A	14CE	
1D1A	14CE	REM *** MIL ONLY PROFILE ***
1D1A	14CE	RECOD1\$ = RECOD1\$ + MILPROS
1D26	14CE	
1D26	14CE	REM *** NOT AVAILABLE ***
1D26	14CE	RECOD1\$ = RECOD1\$ + NAVAIL\$
1D32	14CE	
1D32	14CE	REM *** ADMITTED ***
1D32	14CE	RECOD1\$ = RECOD1\$ + ADMITS
1D3E	14CE	
1D3E	14CE	REM *** INFIELD ***
1D3E	14CE	RECOD1\$ = RECOD1\$ + " " 'NO FIELD ON TMC
1D4A	14CE	
1D4A	14CE	REM *** INJURY ***
1D4A	14CE	RECOD1\$ = RECOD1\$ + INJURS
1D56	14CE	
1D56	14CE	REM *** PURPOSE OF VISIT ***
1D56	14CE	RECOD1\$ = RECOD1\$ + " "
1D62	14CE	
1D62	14CE	REM *** PRIM FOLLOW-UP/RULE OUT ***
1D62	14CE	RECOD1\$ = RECOD1\$ + FU.ROS
1D6E	14CE	

Offset	Data	Source Line
1D6E	14CE	REM *** PRIMARY DX CODE ***
1D6E	14CE	RECOO1\$ = RECOO1\$ + PRIMDX\$
1D7A	14CE	
1D7A	14CE	RETURN
1D7D	14CE	
1D7D	14CE	REM *****
1D7D	14CE	REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
1D7D	14CE	REM *****
1D7D	14CE	280 PAD=MAXLENGTH - LEN(RECOUT\$) 'FIND OUT HOW SHORT THE RECORD IS
1D8B	14D0	RECOU\$ = RECOU\$ + STRING\$(PAD,PAD\$) 'PAD THE RECORD WITH FILL CHAR
1D9E	14D0	RETURN
1DA1	14D0	
1DA1	14D0	299 REM
1DA2	14D0	
1DA2	14D0	REM -----END OF MODULE RACM800.M02-----
1DA2	14D0	
1DA2	14D0	998 REM CONTINUE
1DA3	14D0	
1DA3	14D0	999 READTYPE = 2
1DAA	14D0	IF LNLCOUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
1DBA	14D2	GOTO 10
1DBE	14D2	
1DBE	14D2	REM END OF SCAN/DECODE/WRITE LOOP =====
1DBE	14D2	
1DBE	14D2	1000 REM \$INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
1DBF	14D2	REM *****
1DBF	14D2	REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
1DBF	14D2	REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
1DBF	14D2	REM *****
1DBF	14D2	REM PATIENT QMR INPUT PROGRAM *
1DBF	14D2	REM *
1DBF	14D2	REM This program verifies user is logged on properly. If there is no *
1DBF	14D2	REM valid user logged on at the time of execution, this subroutine will*
1DBF	14D2	REM chain to the logon program RACP05, otherwise a return is issued. *
1DBF	14D2	REM *****
1DBF	14D2	REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
1DBF	14D2	REM *****
1DBF	14D2	1001 OPEN "I",1,"RACLOG.DAT"
1D01	14D2	IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
1D0F	14D2	INPUT #1,USER\$(1),DT\$,TM\$,PID\$
1E00	14DE	IF USER\$(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
1E0E	14DE	IF USER\$(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
1E1C	14DE	CLOSE 1
1E23	14DE	SCREEN 0,1,0,0
1E39	14DE	COLOR FORE,BACK,BORD
1E4F	14DE	CLS
1E53	14DE	RETURN
1E56	14DE	
1E56	14DE	1002 CLOSE
1E5A	14DE	CHAIN "RACP05"
1E61	14DE	'=====END OF LOGON VERIFY SUBROUTINE 1000=====
1E61	14DE	
1E61	14DE	2000 REM \$INCLUDE: 'RACS2000.SUB' INCLUDE THE RIPLY/DELAY SUB
1E62	14DE	REM *****

Offset	Data	Source Line
1E62	14DE	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1E62	14DE	REM **** SKIP COLE ****
1E62	14DE	REM **** SUBROUTINE NAME : RACS2000.SUB ****
1E62	14DE	REM **** SCANNER PROGRAM # : ALL ****
1E62	14DE	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1E62	14DE	REM **** SERVERS AS A WAIT AND REPLY ****
1E62	14DE	REM **** ENTRY MODULE ****
1E62	14DE	REM **** INPUT : SINGLE KEYBOARD ENTRY ****
1E62	14DE	REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
1E62	14DE	REM **** RESERVED LINE ****
1E62	14DE	REM **** NUMBERS : 2001-2010 ****
1E62	14DE	REM *****
1E62	14DE	2001 REM REPLY FUNCTION
1E63	14DE	2002 REPLY\$=INKEY\$: IF REPLY\$="" THEN 2002
1E77	14E2	REPLY=ASC(REPLY\$)
1E81	14E4	IF REPLY > 90 THEN REPLY\$=CHR\$(REPLY XOR 32) 'CONVERT TO CAPS
1E9C	14E4	IF REPLY\$ < "A" OR REPLY\$ > "Z" THEN REPLY\$="?"
1EC8	14E4	RETURN
1EC8	14E4	
1EC8	14E4	5000 REM \$INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB
1ECC	14E4	REM *****
1ECC	14E4	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1ECC	14E4	REM **** SKIP COLE ****
1ECC	14E4	REM **** SUBROUTINE NAME : RXXS5000.SUB ****
1ECC	14E4	REM **** SCANNER PROGRAM # : ALL ****
1ECC	14E4	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1ECC	14E4	REM **** PERFORMS A DATE EDIT ****
1ECC	14E4	REM **** INPUT : DATE TO BE CHECKED MUST BE ****
1ECC	14E4	REM **** IN THE VARIABLE NAMED ****
1ECC	14E4	REM **** 'CK.5000\$' ****
1ECC	14E4	REM **** IN THE FORMAT "YYMMDD" ****
1ECC	14E4	REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
1ECC	14E4	REM **** VARIABLE. IF THIS VARIABLE ****
1ECC	14E4	REM **** CONTAINS ANY NUMBER OTHER ****
1ECC	14E4	REM **** THAN 0, AN ERROR WAS FOUND ****
1ECC	14E4	REM **** IN THE DATE. ****
1ECC	14E4	REM **** RESERVED LINE ****
1ECC	14E4	REM **** NUMBERS : 5001-5009 ****
1ECC	14E4	REM *****
1ECC	14E4	RT.5000 = 0
1ED3	14E4	CKYEAR = VAL(LEFT\$(CK.5000\$,2)) 'YEAR NUMERIC VALUE
1EE6	14E6	CKMONTH = VAL(MID\$(CK.5000\$,3,2)) 'MONTH NUMERIC VALUE
1EFC	14E8	CKDAY = VAL(RIGHT\$(CK.5000\$,2)) 'DAY NUMERIC VALUE
1F0F	14EA	
1F0F	14EA	IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
1F25	14EA	IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
1F38	14EA	IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
1F51	14EA	IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
1.67	14FA	

Offset	Data	Source Line
1F67	14EA	REM LEAP YEAR CHECK
1F67	14EA	MOLENGTH(2) = 28
1F6E	14EA	IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY
1F7D	14EA	IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR
1F92	14EA	MOLENGTH(2) = 29
1F99	14EA	
1F99	14EA	5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
1FB8	14EA	
1FB8	14EA	5009 RETURN
1FB8	14EA	
1FB8	14EA	REM -----END OF SUBROUTINE 5000 -----
1FB8	14EA	
1FB8	14EA	
1FB8	14EA	5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTP UCA VALIDATE SUB
1FBC	14EA	5600 REM INCLUDE: 'RACS5600.SUB' INCLUDE THE INP UCA VALIDATE SUB
1FBD	14EA	5700 REM \$INCLUDE: 'RACS5700.SUB' INCLUDE MAP ONES TO POSITION NO.
1FBE	14EA	REM *****
1FBE	14EA	REM **** AMBULATORY CARE DATA BASE 29 JUL 85 ****
1FBE	14EA	REM **** D R BOLLING ****
1FBE	14EA	REM **** SUBROUTINE NAME : RXXS5700.SUB ****
1FBE	14EA	REM **** SCANNER PROGRAM # : ALL ****
1FBE	14EA	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1FBE	14EA	CONVERTS A BINARY ARRAY INTO ****
1FBE	14EA	TWO CHAR CODES. ****
1FBE	14EA	REM ****
1FBE	14EA	REM **** INPUT : INP.STO\$ AS STRING ****
1FBE	14EA	REM ****
1FBE	14EA	REM ****
1FBE	14EA	REM **** OUTPUT : BUF.STO\$ AS STRING. ****
1FBE	14EA	REM ****
1FBE	14EA	REM ****
1FBE	14EA	REM **** RESERVED LINE ****
1FBE	14EA	REM **** NUMBERS : 5710-5730 ****
1FBE	14EA	REM *****
1FBE	14EA	
1FBE	14EA	BUF.STO\$=""
1FC7	14EA	N.STO=1
1FCE	14EA	5710 X.STO=INSTR(N.STO,INP.STO\$,"1")
1FE0	14EA	IF X.STO=0 THEN GOTO 5720 'THATS ALL
1FEF	14EA	N.STO = X.STO + 1 'NEXT STARTING POINT
1FF7	14EA	X.STO = X.STO + 100 'PAD WITH LEADING ZERO
2001	14EA	BUF.STO\$ = BUF.STO\$ + RIGHT\$(STR\$(X.STO),2)
2017	14EA	IF N.STO <= LEN(INP.STO\$) THEN GOTO 5710
202A	14EA	
202A	14EA	5720 RETURN
202D	14EA	
202D	14EA	REM ----- END OF SUBROUTINE 5700 -----
202D	14EA	5800 REM \$INCLUDE: 'RACS5800.SUB' INCLUDE FOUR COL DX CONVERTER
202E	14EA	REM *****
202E	14EA	REM *** AMBULATORY CARE DATA BASE 30 JUL 85 ***
202E	14EA	REM *** D R BOLLING ***
202E	14EA	REM *** SUBROUTINE NAME : RXXS5800.SUB ***
202E	14EA	REM *** SCANNER PROGRAM : AS APPROPRIATE ***
202E	14EA	REM *** FUNCTION : THIS SUBROUTINE MODULE READS ***

Offset	Data	Source Line
202E	14EA	REM *** FOUR COLUMNS IN DX AREA AND ***
202E	14EA	REM *** CONVERTS TO A POSITION IN A ***
202E	14EA	REM *** TABLE. AN ERROR CODE IS ***
202E	14EA	REM *** RETURNED IF MULTIPLE CODES ARE ***
202E	14EA	REM *** PRESENT. ***
202E	14EA	REM ***
202E	14EA	REM *** INPUT : X.POS - STARTING POSITION IN ***
202E	14EA	REM *** STRING ***
202E	14EA	REM ***
202E	14EA	REM *** OUTPUT : X.FIN - TABLE POSITION OF DX ***
202E	14EA	REM *** RT.5800 = 1 ON ERROR ***
202E	14EA	REM ***
202E	14EA	REM *** RESERVED LINE NUMBERS : 5801 - 5899 ***
202E	14EA	REM *****
202E	14EA	RT.5800=0 'INITIALIZE ERROR INDICATOR
2035	14EA	X.FIN=0 'INITIALIZE TABLE OFFSET RESULT
203C	14EA	
203C	14EA	REM ** COLUMN 1 **
203C	14EA	X=0 'STARTING POINTER
2043	14EA	X.SIZ=C1SIZ
204A	14EC	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
2061	14EC	IF X.STO=0 THEN GOTO 5802
2070	14EC	X.FIN=X.STO
2077	14EC	
2077	14EC	5802 REM ** COLUMN 2 **
2078	14EC	IF C2SIZ=0 THEN GOTO 5804
2087	14EC	X=X+X.SIZ
2092	14EC	X.POS=X.POS+2
209B	14EC	X.SIZ=C2SIZ
20A2	14EC	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
20B9	14EC	IF X.STO=0 THEN GOTO 5804
20C8	14EC	IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
20D7	14EC	X.FIN=X.STO + X
20E2	14EC	
20E2	14EC	5804 REM ** COLUMN 3 **
20E3	14EC	IF C3SIZ=0 THEN GOTO 5806
20F2	14EC	X=X+X.SIZ
20FD	14EC	X.POS=X.POS+2
2106	14EC	X.SIZ=C3SIZ
210D	14EC	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
2124	14EC	IF X.STO=0 THEN GOTO 5806
2133	14EC	IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
2142	14EC	X.FIN=X.STO + X
214D	14EC	
214D	14EC	5806 REM ** COLUMN 4 **
214E	14EC	IF C4SIZ=0 THEN GOTO 5808
215D	14EC	X=X+X.SIZ
2168	14EC	X.POS=X.POS+2
2171	14EC	X.SIZ=C4SIZ
2178	14EC	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
218F	14EC	IF X.STO=0 THEN GOTO 5808
219E	14EC	IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
21AD	14EC	X.FIN=X.STO + X
21B8	14EC	

PAGE 25
05-26-87
09:33:12

```

Offset Data Source Line
2188 14EC 5808 REM
2189 14EC GOTO 5899
218D 14EC
218D 14EC 5890 RT.5800=1 'ERROR - MULTIPLE CODES
21C4 14EC
21C4 14EC 5899 RETURN
21C7 14EC
21C7 14EC REM -----END OF SUBROUTINE RXXS5800.SUB-----
21C7 14EC
21C7 14EC
21C7 14EC 6000 REM $INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE SUB
21C8 14EC REM *****
21C8 14EC REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
21C8 14EC REM **** SKIP COLE ****
21C8 14EC REM **** SUBROUTINE NAME : RXXS6000.SUB ****
21C8 14EC REM **** SCANNER PROGRAM # : ALL ****
21C8 14EC REM **** FUNCTION : THIS SUBROUTINE MODULE ****
21C8 14EC REM **** PERFORMS INSTRING SEARCH ****
21C8 14EC REM ****
21C8 14EC REM **** INPUT : STRING TO BE SEARCHED MUST ****
21C8 14EC REM **** BE IN THE VARIABLE NAMED : ****
21C8 14EC REM **** 'X$' ****
21C8 14EC REM ****
21C8 14EC REM **** OUTPUT : 'TOT' = TOTAL NUMBER OF ****
21C8 14EC REM **** HITS IN THE DESTRING ****
21C8 14EC REM **** 'HOLD$()' IS THE ARRAY ****
21C8 14EC REM **** CONTAINING THE NUMERIC ****
21C8 14EC REM **** VALUE OF THE HIT POSITIONS ****
21C8 14EC REM ****
21C8 14EC REM *****
21C8 14EC REM **** RESERVED LINE ****
21C8 14EC REM **** NUMBERS : 6001-6009 ****
21C8 14EC REM *****
21C8 14EC 6001 PTR = INSTR('1',1)
21D6 14EE TOT = 0
21D0 14F0 WHILE PTR > 0
21E8 14F0 TOT=TOT+1
21F0 14F0 HOLD$(TOT) = RIGHT$(STR$(PTR),2)
2212 14F0 PTR=PTR+1
221A 14F0 PTR = INSTR(PTR,X$,1)
222C 14F0 WEND
2230 14F0 RETURN
2233 14F0
2233 14F0 REM -----END OF SUBROUTINE RXXS6000.SUB-----
2233 14F0
2233 14F0 7000 REM $INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
2234 14F0 REM *****
2234 14F0 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
2234 14F0 REM **** SKIP COLE ****
2234 14F0 REM **** SUBROUTINE NAME : RACS7000.SUB ****
2234 14F0 REM **** SCANNER PROGRAM # : ALL ****
2234 14F0 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
2234 14F0 REM **** PRINTS THE STANDARD SCREEN ****

```

```

Offset  Data  Source Line
2234  14F0  REM  ****      HEADING.      ****
2234  14F0  REM  ****      INPUT      :  COMMON VARIABLE USERS(2)  ****
2234  14F0  REM  ****      SYSTEM DATE      ****
2234  14F0  REM  ****      OUTPUT      :  SCREEN HEADING      ****
2234  14F0  REM  ****      RESERVED LINE      ****
2234  14F0  REM  ****      NUMBERS      :  7001-7010      ****
2234  14F0  REM  ****
2234  14F0  7001  LOCATE 1,1
223E  14F0      PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
2246  14F0      LOCATE 1,65
2253  14F0      PRINT DATES;
2258  14F0      LOCATE 2,1
2268  14F0      PRINT "USER : ";USERS(1)
2275  14F0      RETURN
2278  14F0
2278  14F0  7100  REM $INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
2279  14F0  REM  ****
2279  14F0  REM  ****      AMBULATORY CARE DATA BASE      13 APR 85      ****
2279  14F0  REM  ****      SKIP COLE      ****
2279  14F0  REM  ****      SUBROUTINE NAME      :  RXXS7100.SUB      ****
2279  14F0  REM  ****      SCANNER PROGRAM #      :  ALL      ****
2279  14F0  REM  ****      FUNCTION      :  THIS SUBROUTINE MODULE      ****
2279  14F0  REM  ****      PRINTS THE STANDARD HEADING      ****
2279  14F0  REM  ****      ON THE PRINTER.      ****
2279  14F0  REM  ****      INPUT      :  DATE,PAGE,PGMID$,PGMTITL$      ****
2279  14F0  REM  ****      OUTPUT      :  PRINTER HEADING, LN.COUNT      ****
2279  14F0  REM  ****      RESERVED LINE      ****
2279  14F0  REM  ****      NUMBERS      :  7101-7110      ****
2279  14F0  REM  ****
2279  14F0  7101  IF PAGE > 0 THEN LPRINT CHR$(12);
228F  14F0      LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITL$;
229C  14F4      LPRINT TAB(70);DATES
22AF  14F4      PAGE=PAGE+1
22B7  14F4      LPRINT "PROGRAM ";PGMID$;TAB(70);"PAGE";
22D4  14F4      LPRINT USING "####";PAGE
22E0  14F4      LPRINT
22E8  14F4      LN.COUNT=3
22EF  14F4      RETURN
22F2  14F4
22F2  14F4  8000  REM $INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
22F3  14F4  REM  ****
22F3  14F4  REM  ****      AMBULATORY CARE DATA BASE      13 APR 85      ***
22F3  14F4  REM  ****      SKIP COLE      ***
22F3  14F4  REM  ****      SUBROUTINE NAME      :  RXXS8000.SUB      ***
22F3  14F4  REM  ****      SCANNER PROGRAM #      :  ALL      ***
22F3  14F4  REM  ****      FUNCTION      :  THIS SUBROUTINE MODULE      ***
22F3  14F4  REM  ****      IS A GROUPING THAT PERFORMS      ***
22F3  14F4  REM  ****      VARIOUS DECODING FUNCTIONS      ***

```

Offset	Data	Source Line
22F3	14F4	REM ***** ON THE SCANNER DATA ***
22F3	14F4	REM ***** ***
22F3	14F4	REM ***** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
22F3	14F4	REM ***** 8050 - CHECK FOR END OF JOB ***
22F3	14F4	REM ***** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
22F3	14F4	REM ***** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
22F3	14F4	REM ***** (RETURNED IN TEXT\$ STRING VARIABLE) ***
22F3	14F4	REM ***** ***
22F3	14F4	REM ***** INPUT : SHEET RECORD, RECORD LENGTH ***
22F3	14F4	REM ***** ***
22F3	14F4	REM ***** OUTPUT : 'TEXT\$' TRING VARIABLE ***
22F3	14F4	REM ***** ***
22F3	14F4	REM ***** RESERVED LINE ***
22F3	14F4	REM ***** NUMBERS : 8001-8500 ***
22F3	14F4	REM *****
22F3	14F4	
22F3	14F4	'DECODE THE HEADER ONLY
22F3	14F4	8001 POINTIF 0
22FA	14F6	RECORDPTR = VARPTR(SHEETREC(0))
2301	14F8	FOR J8000 = 1 TO 21
2308	14F8	8002 TEXT\$= TEXT\$+CHR\$(PEEK(RECORDPTR + POINTER))
2326	14F8	POINTER=POINTER+1
232E	14F8	NEXT J8000
2330	14FA	PROGRAM\$= LEFT\$(TEXT\$,3)
234C	14FE	BATCH\$= MID\$(TEXT\$,4,3)
235E	1502	SERIAL\$= MID\$(TEXT\$,7,4)
2370	1506	RUNID\$= MID\$(TEXT\$,11,1)
2382	150A	FORM\$= MID\$(TEXT\$,12,2)
2394	150E	POCKET\$= MID\$(TEXT\$,14,1)
23A6	1512	SCANERR1\$=MID\$(TEXT\$,16,2)
23B8	1516	SCANERR2\$=MID\$(TEXT\$,18,2)
23CA	151A	SCANERR3\$=MID\$(TEXT\$,20,2)
23DC	151E	GOTO 8500
23E0	151E	
23E0	151E	8050 REM CHECK FOR END OF JOB/END OF BATCH
23E1	151E	IF PROGRAM\$ = PGMID\$ THEN GOTO 8500
23F3	151E	LPRINT STRING\$(80,"")
2401	151E	LPRINT
2409	151E	LPRINT "RECORDS PROCESSED ... ";SERIAL\$
2416	151E	LPRINT "STARTED AT ";BTIM\$
2423	151E	LPRINT "ENDED AT ";TIM\$
2430	151E	LPRINT CHR\$(12)
243B	151E	GOTO 30000
243F	151E	
243F	151E	8070 REM CHECK FOR SCANNER ERRORS
2440	151E	IF POCKET\$ = " " GOTO 8500
2452	151E	LPRINT LITHO\$;
245A	151E	LPRINT " ... SCANNER ERRORS : ";
2462	151E	LPRINT SCANERR1\$;" / ";
246F	151E	LPRINT SCANERR2\$;" / ";
247C	151E	LPRINT SCANERR3\$
2484	151E	LN=LN+1
248C	1520	GOTO 999
2490	1520	

```

Offset  Data   Source Line
-----  -
2490  1520    8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
2491  1520          LOCATE 5,1:PRINT "PROGRAM ";PROGRAMS;
24AB  1520          PRINT " BATCH ";BATCHS;
24B8  1520          PRINT "  RUN ";RUNIDS;
24C5  1520          PRINT "  FORM ";FORMS;
24D2  1520          PRINT " POCKET ";POCKET$
24DF  1520          GOTO 8500
24E3  1520
24E3  1520    8200 REM DECODE THE RESPONSE POSITIONS
24E4  1520          POINTER = 21
24EB  1520          RECORDPTR = VARPTR(SHEETREC(0))
24F2  1520          FOR J8000 = 22 TO RECORDLENGTH
24FF  1522    8202          TEXT$ = TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
251D  1522          POINTER=POINTER+1
2525  1522          NEXT J8000
2536  1522
2536  1522    8500 RETURN
2539  1522
2539  1522          REM ----- END OF RXXS8000.SUB -----
2539  1522
2539  1522    9000 REM $INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
253A  1522 REM *****
253A  1522 REM *****  AMBULATORY CARE DATA BASE          13 APR 85      *****
253A  1522 REM *****                                SKIP COLE      *****
253A  1522 REM *****  PROGRAM NAME      :  RACS9000.SUB          *****
253A  1522 REM *****  SCANNER PROGRAM #    :  ALL                  *****
253A  1522 REM *****  FUNCTION      :  THIS SUBROUTINE MODULE      *****
253A  1522 REM *****                                CONTROLS THE SCANNER I/O *****
253A  1522 REM *****
253A  1522 REM *****  INPUT/OUTPUT      :  REFER TO THE ASYNCHRONOUS *****
253A  1522 REM *****                                COMMUNICATIONS MANUAL AND THE *****
253A  1522 REM *****                                PRE-RELEASED SOFTWARE GUIDE *****
253A  1522 REM *****
253A  1522 REM *****
253A  1522 REM *****  RESERVED LINE *****
253A  1522 REM *****  NUMBERS      :  9001-9100 *****
253A  1522 REM *****
253A  1522
253A  1522
253A  1522 REM *****
253A  1522 REM ***** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER *****
253A  1522 REM ***** ARGUMENTS: PRESET ... SEE BELOW *****
253A  1522 REM *****
253A  1522    9001 REM
253B  1522          PROTOCOL(0) = 9600          'BAUD RATE
2542  1522          PROTOCOL(1) = 78          'PARITY (SEE PAGE 4-8 OF MANUAL)
2549  1522          PROTOCOL(2) = 8          'DATA BITS
2550  1522          PROTOCOL(3) = 1          'STOP BITS
2557  1522          PROTOCOL(4) = 2          'RS-232 PORT
255E  1522          PROTOCOL(5) = 0          'WRITE TIME-OUT
2565  1522          PROTOCOL(6) = 0          'READ TIME-OUT
256C  1522
256C  1522          ERRSTAT$ = SPACE$(60)

```

Offset	Data	Source Line
2578	1522	ARGPTR = VARPTR(PROTOCOL(0))
257F	1524	CALL SETUP (ARGPTR,ERRSTAT\$)
2590	1524	ERRMSG\$=""
2599	1524	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="SETUP ERROR "+ERRSTAT\$
25B5	1524	GOTO 9100
25B9	1524	
25B9	1524	REM *****
25B9	1524	REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
25B9	1524	REM **** ARGUMENTS: CNTRLOPT ****
25B9	1524	REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
25B9	1524	REM **** CNTRLOPT = 2 = STOP SCANNER (S0) ****
25B9	1524	REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
25B9	1524	REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
25B9	1524	REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
25B9	1524	REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
25B9	1524	REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
25B9	1524	REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
25B9	1524	REM *****
25B9	1524	9010 REM
25BA	1524	ERRSTAT\$ = SPACE\$(60)
25C6	1524	CALL CNTRL (CNTRLOPT,ERRSTAT\$)
25D7	1524	ERRMSG\$=""
25E0	1524	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
25FC	1524	GOTO 9100
2600	1524	
2600	1524	REM *****
2600	1524	REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
2600	1524	REM ****
2600	1524	REM **** ARGUMENTS: READTYPE ****
2600	1524	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
2600	1524	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
2600	1524	REM ****
2600	1524	REM **** ARGUMENTS: RECORDLENGTH ****
2600	1524	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
2600	1524	REM **** TRANSMITTED ****
2600	1524	REM *****
2600	1524	9020 REM
2601	1524	ERRSTAT\$ = SPACE\$(60)
2600	1524	RECORDPTR = VARPTR(SHEETREC(0))
2614	1524	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
2620	1524	ERRMSG\$=""
2636	1524	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
2657	1524	GOTO 9100
2658	1524	
2658	1524	REM *****
2658	1524	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
2658	1524	REM ****
2658	1524	REM **** ARGUMENTS: PRINTPOS ****
2658	1524	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
2658	1524	REM **** VALUES = 0 THRU 90 ****
2658	1524	REM ****
2658	1524	REM **** ARGUMENTS: PSTRING\$ ****
2658	1524	REM **** TEXT TO BE PRINTED ON THE FORM ****
2658	1524	REM ****

Offset	Data	Source Line
265B	1524	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
265B	1524	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
265B	1524	REM ****
265B	1524	9030 REM
265C	1524	ERRSTAT\$ = SPACES(60)
2668	1524	RECORDPTR = VARPTR(SHEETREC(0))
266F	1524	CALL TPRINT(PRINTPOS,PSTRING\$,ERRSTAT\$)
2684	1524	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
26A0	1524	GOTO 9100
26A4	1524	
26A4	1524	9100 RETURN
26A7	1524	REM -----END OF SUBROUTINE RACS9000.SUB -----
26A7	1524	
26A7	1524	REM END OF SUBROUTINES =====
26A7	1524	
26A7	1524	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
26A8	1524	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
26B0	1524	LPRINT "ERASING FILE ";DATFIL\$
26B0	1524	BEEP
26C1	1524	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
26CD	1524	CLOSE
26D1	1524	OPEN DATFIL\$ FOR OUTPUT AS #1
26E3	1524	PRINT #1,STRING\$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
26F5	1524	CLOSE
26F9	1524	
26F9	1524	30000 REM
26FA	1524	CLOSE
26FE	1524	CHAIN "RACP10"
2705	1524	
2708	1524	

22151 Bytes Available

1'662 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

PAGE 1

04-29-87

08:1E:35

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```
001A 0002 REM $LINESIZE: 132
001A 0002 REM $PAGESIZE: 66
001A 0002 REM $TITLE: 'RACP820 '
001A 0002 REM $SUBTITLE: 'EKG (DDAA) FORM'
001A 0002 REM $PAGE
```

```

Offset  Data  Source Line
-----
001A 0002  REM +-----+
001A 0002  REM | NAME: RACP820          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002  REM | DATE: 14 MAR 87       EKG (DDAA) VISIT |
001A 0002  REM | D R BOLLING          SHORT FORM |
001A 0002  REM |
001A 0002  REM | INCLUDES PREFIX TO LITHO FOR EACH PATIENT ON FORM |
001A 0002  REM +-----+
001A 0002  REM                      EKG FORM INPUT PROGRAM
001A 0002  REM
001A 0002  REM This program reads the SHORT form OMR data, converts various
001A 0002  REM fields, prints an error report and produces the file:
001A 0002  REM
001A 0002  REM                      VISIT.DAT
001A 0002  REM
001A 0002  REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002  REM each time the program is run. Thus, if the file does not exist,
001A 0002  REM records will be added to the front. If the file exists, records
001A 0002  REM will be added to the end of the current file. It is intended that
001A 0002  REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002  REM the data file after the load has been successfully accomplished.
001A 0002  REM
001A 0002  REM
001A 0002  REM If there is no valid user logged at the time of execution, this
001A 0002  REM program will chain to the logon program RACP05, otherwise,
001A 0002  REM THE PROGRAM CHAINS TO PROGRAM RACP10 ON EXIT.
001A 0002  REM
001A 0002  REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002  REM *****
001A 0002  REM * NAME: RACDIM.MOD          DIMENSION DEFINITIONS *
001A 0002  REM * Date: 28 Feb 84          Written by: Floyd Cole *
001A 0002  REM *****
001A 0002  REM ' Dimensioned variables are defined in this file.
001A 0002  REM ' It is an included file so it cannot be run in a stand-alone,
001A 0002  REM ' mode.
001A 0002  REM '
001A 0002  REM ' This program segment may be modified, but all files containing
001A 0002  REM ' an include for this segment must be re-compiled in order to
001A 0002  REM ' affect the changes made here.
001A 0002  REM ' ***** START OF DIMENSION DEFINITION *****
001A 0002  REM
001A 0002  REM DEFINIT A-Z
001A 0002  REM DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002  REM
001A 0002  REM ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002  REM
001A 0002  REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002  REM
001A 0002  REM DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002  REM DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002  REM DIM ED.MSG$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002  REM DIM CLINIC2.PFX$(5) '(PREFIX -A B D F S- FOR CLINIC #2)
001A 0002  REM DIM PROCED$(18) '(PROCEDURE TABLE FOR SHORT FORM)
001A 0002  REM DIM PROVIDER.TIME$(9) '(PROVIDER TIME TABLE)

```

Offset	Data	Source Line
001A	0002	DIM REFCOD\$(12) '(REFERRAL CODE TABLE)
001A	0002	DIM CLIBUB\$(6) '(OTHER UCA)
001A	0002	DIM SPECOD\$(9) '(SPECIFIC PREASSIGNED CLINIC CODES)
001A	0002	DIM SSN.COD\$(4) ' .
001A	0002	DIM FMP.COD\$(4) ' .
001A	0002	DIM REF.COD\$(4) ' .
001A	0002	DIM PR1.TIM\$(4) ' . ARRAYS TO HOLD MULTIPLE VISITS
001A	0002	DIM TOT.PROC(4) ' .
001A	0002	DIM SPE.COD\$(4) ' .
001A	0002	DIM TOT.SPE(4) ' .
001A	0002	DIM GROUP1\$(4,19) '(PATIENT/PROCEDURE GROUP PROV 1)
001A	0002	DIM HOLDS(18) '(HOLD AREA FOR SUBROUTINE 6000)
001A	0002	
001A	0002	REM \$INCLUDE: 'RACCMN.MOD' REM INCLUDE THE COMMON AREA DEFINITION
001A	0002	*****
001A	0002	* NAME: RACCMN.MOD COMMON AREA DEFINITION *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A	0002	' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	' *****START OF COMMON DEFINITIONS*****
001A	0002	
001A	0002	COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL\$ 'BASIC SCREEN COLORS
001A	0002	COMMON HEADERS\$ '21 CHARACTER SCANNER HEADER INFO
001A	0002	COMMON TEXT\$ ' AINING CHARACTERS FROM SCANNER
001A	0002	COMMON PGMID\$ 'PROGRAM OR FORM ID
001A	0002	COMMON MOLENGTH() 'DAYS IN THE MONTH
001A	0002	COMMON USERS()
001A	0002	' *****END OF COMMON DEFINITION*****
001A	0002	
001A	0002	
001A	0002	REM \$INCLUDE: 'RACDEF.MOD' REM INCLUDE THE DEFAULT DEFINITIONS
001A	0002	*****
001A	0002	* NAME: RACP01.DEF DEFAULT DEFINITIONS *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' Variables used in common that have a default value on start*up
001A	0002	' will be held in this file. It is an included file so it cannot
001A	0002	' be run in a stand*alone mode. In normal operation, this file
001A	0002	' should be 'included' in the main program only (RACP10.BAS).
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	' *****START OF DEFAULT DEFINITION*****
001A	0002	FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
004F	11A4	BACK = 1 'Background Color = Light Blue

Offset	Data	Source Line
0056	11A4	BORD = 4 'BORDER = RED
005D	11A6	HIDE = 4 'ALTERNATE COLOR = RED
0064	11A6	EFORE= 14 'ERROR FOREGROUND DISPLAY
006B	11A6	EBACK= 0 'ERROR BACKGROUND DISPLAY
0072	11A6	BELL\$ = CHR\$(7) 'Sound the bell
007E	11A6	
007E	11A6	MOLENGTH(1) = 31 'JAN
0085	11A6	MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
008C	11A6	MOLENGTH(3) = 31 'MAR
0093	11A6	MOLENGTH(4) = 30 'APR
009A	11A6	MOLENGTH(5) = 31 'MAY
00A1	11A6	MOLENGTH(6) = 30 'JUN
00A8	11A6	MOLENGTH(7) = 31 'JUL
00AF	11A6	MOLENGTH(8) = 31 'AUG
00B6	11A6	MOLENGTH(9) = 30 'SEP
00BD	11A6	MOLENGTH(10) = 31 'OCT
00C4	11A6	MOLENGTH(11) = 30 'NOV
00CB	11A6	MOLENGTH(12) = 31 'DEC
00D2	11A6	
00D2	11A6	DATEERR\$(0) = " "
00DB	11A6	DATEERR\$(1) = "INVALID MONTH"
00E4	11A6	DATEERR\$(2) = "INVALID DAY "
00ED	11A6	DATEERR\$(3) = "DAY TOO LARGE FOR MONTH CODED"
00F6	11A6	
00F6	11A6	MAXLENGTH = 80 'MAXIMUM LENGTH OF OUTPUT RECORD
00FD	11A8	PAD\$ = ", " 'PAD CHARACTER FOR SHORT RECORDS
0106	11AC	
0106	11AC	' *****END OF DEFAULT DEFINITION*****
0106	11AC	
0106	11AC	KEY OFF
010C	11AC	
010C	11AC	REM *****
010C	11AC	REM THE FOLLOWING VARIABLES ARE UNIQUE TO EACH PROGRAM AND MUST
010C	11AC	REM BE CHANGED.
010C	11AC	REM *****
010C	11AC	PGMTITL\$ = "EKG (DDAA) FORM"
0115	11B0	
0115	11B0	PGMID\$ = "820" 'VALUE RECEIVED FROM THE SCANNER
011E	11B0	'IN HEADER VARIABLE 'PROGRAM\$'
011E	11B0	
011E	11B0	DATFIL\$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
0127	11B4	
0127	11B4	REM LENGTH OF STRING RECEIVED FROM THE OMR....
0127	11B4	HEADER = 21
012E	11B6	RESPONSE= 218
0135	11B8	RECORDLENGTH = HEADER + RESPONSE
0140	11BA	
0140	11BA	N.PROC = 18 ' NUMBER OF PROCEDURES FOR THIS FORM
0147	11BC	
0147	11BC	REM *****
0147	11BC	
0147	11BC	BTIME%=TIME\$ 'SCAN START TIME
0150	11C0	

RACP820
EKG (DDAA) FORM

PAGE 5
04-29-87
08:18:35

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
0150	11C0	REM *** OTHER UCA TABLE ***
0150	11C0	CLIBUB\$(0)="DDAA" : CLIBUB\$(4)="DDAF"
0162	11C0	CLIBUB\$(1)="DDAB" : CLIBUB\$(5)="DDAJ"
0174	11C0	CLIBUB\$(2)="DDAI" : CLIBUB\$(6)="DDAO"
0186	11C0	CLIBUB\$(3)="DDAN"
018F	11C0	
018F	11C0	REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
018F	11C0	REM CLINIC #2
018F	11C0	CLINIC2.PFX\$(0)=" "
0198	11C0	CLINIC2.PFX\$(1)="A"
01A1	11C0	CLINIC2.PFX\$(2)="B"
01AA	11C0	CLINIC2.PFX\$(3)="D"
01B3	11C0	CLINIC2.PFX\$(4)="F"
01BC	11C0	CLINIC2.PFX\$(5)="S"
01C5	11C0	
01C5	11C0	REM *** PREASSIGNED REFERRAL CODES ***
01C5	11C0	REFCOD\$(1)="AAAA" : REFCOD\$(7)="BGYA"
01D7	11C0	REFCOD\$(2)="ABAA" : REFCOD\$(8)="BHAA"
01E9	11C0	REFCOD\$(3)="BAAA" : REFCOD\$(9)="BHA"
01FB	11C0	REFCOD\$(4)="BACA" : REFCOD\$(10)="BIYA"
0200	11C0	REFCOD\$(5)="BAIA" : REFCOD\$(11)="FBAC"
021F	11C0	REFCOD\$(6)="BBAA" : REFCOD\$(12)="FBGA"
0231	11C0	
0231	11C0	REM *** ENCOUNTER FORM PROCEDURE TABLE ***
0231	11C0	PROCED\$(0)=" " : PROCED\$(7) = "93005" : PROCED\$(14)= "93012"
024C	11C0	PROCED\$(1)="93300" : PROCED\$(8) = "93000" : PROCED\$(15)= "93015"
0267	11C0	PROCED\$(2)="93307" : PROCED\$(9) = "93274" : PROCED\$(16)= "93018"
0282	11C0	PROCED\$(3)="93320" : PROCED\$(10) = "93278" : PROCED\$(17)= "78419"
029D	11C0	PROCED\$(4)="93305" : PROCED\$(11) = "93798" : PROCED\$(18)= "90601"
02B8	11C0	PROCED\$(5)="93308" : PROCED\$(12) = "93202"
02CA	11C0	PROCED\$(6)="93010" : PROCED\$(13) = "93201"
02DC	11C0	
02DC	11C0	REM *** PROVIDER TIME TABLE ***
02DC	11C0	PROVIDER.TIMES\$(00)="000" ' NO TIME
02E5	11C0	PROVIDER.TIMES\$(01)="002" ' 2 MINUTES
02EE	11C0	PROVIDER.TIMES\$(02)="005" ' 5 MINUTES
02F7	11C0	PROVIDER.TIMES\$(03)="010" ' 10 MINUTES
0300	11C0	PROVIDER.TIMES\$(04)="015" ' 15 MINUTES
0309	11C0	PROVIDER.TIMES\$(05)="020" ' 20 MINUTES
0312	11C0	PROVIDER.TIMES\$(06)="030" ' 30 MINUTES
0318	11C0	PROVIDER.TIMES\$(07)="040" ' 40 MINUTES
0324	11C0	PROVIDER.TIMES\$(08)="050" ' 50 MINUTES
032D	11C0	PROVIDER.TIMES\$(09)="060" ' 1 HOUR
0336	11C0	
0336	11C0	REM INCLUDE: 'UCABAMC.OPT' INCLUDE OUTPATIENT UCA TABLE
0336	11C0	
0336	11C0	REM \$PAGE

Offset	Data	Source Line
0336	11C0	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
033B	11C0	CLS
033F	11C0	GOSUB 7000 'PRINT SCREEN HEADING
0344	11C0	
0344	11C0	REM *****
0344	11C0	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
0344	11C0	REM *****
0344	11C0	REM
0344	11C0	OPEN DATFIL\$ FOR APPEND AS #1
0356	11C0	
0356	11C0	REM *****
0356	11C0	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
0356	11C0	REM *****
0356	11C0	LPRINT CHR\$(15);
0361	11C0	WIDTH "LPT1:",160
036B	11C0	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
0377	11C2	COLOR 14
037E	11C2	LOCATE 11,26 : PRINT "EKG FORM "
0393	11C2	COLOR FORE,BACK,BORD
03A9	11C2	
03A9	11C2	REM *****
03A9	11C2	REM **** COMMUNICATIONS SETUP ****
03A9	11C2	REM *****
03A9	11C2	REM PROTOCOL
03A9	11C2	GOSUB 9001
03AE	11C2	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
03C8	11C6	
03C8	11C6	REM START SCANNER (S1)
03C8	11C6	CNTRLOPT =1 :GOSUB 9010
03D4	11C8	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
03EE	11C8	
03EE	11C8	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
0403	11C8	READTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
040A	11CA	
040A	11CA	REM *****
040A	11CA	REM **** SET SCAN SHEET CALL ****
040A	11CA	REM *****
040A	11CA	REM
040A	11CA	
040A	11CA	10 REM - RETURN POINT TO READ NEXT SHEET
040B	11CA	
040B	11CA	AS=INKEY\$
0414	11CE	IF AS=CHR\$(27) THEN GOTO 25000
042A	11CE	
042A	11CE	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
042F	11CE	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
044B	11D2	
044B	11D2	TEXT\$="" 'CLEAR THE INPUT AREA
0454	11D2	GOSUB 8000 'DECODE HEADER
0459	11D2	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
045E	11D2	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
0463	11D2	LITHO\$ = MID\$(TEXT\$,22,8)
0475	11D6	GOSUB 8070 'CHECK FOR SCANNER ERRORS
047A	11D6	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

Offset Data Source Line

```

047F 11D6
047F 11D6 REM $INCLUDE: 'RACM820.MO1' INCLUDE THE SHORT FORM REFORMAT/EDIT MOD
047F 11D6 REM *****
047F 11D6 REM **** AMBULATORY CARE INFORMATION SYSTEM 14 MAR 87 ****
047F 11D6 REM **** D R BOLLING ****
047F 11D6 REM **** MODULE NAME : RACM820.MO1 ****
047F 11D6 REM **** SCANNER PROGRAM # : 820- EKG (DDAA) FORM ****
047F 11D6 REM ****
047F 11D6 REM **** PURPOSE : REFORMAT/EDIT THE ENCOUNTER ****
047F 11D6 REM **** SHORT FORM OMR RECORD. ****
047F 11D6 REM *****
047F 11D6 REM **** RESERVED LINE NUMBERS 100-199 ****
047F 11D6 REM *****
047F 11D6
047F 11D6 N.ERR =0 'COUNTS THE NUMBER OF ERRORS
0486 11D8
0486 11D8 REM *** LITHO CODE DONE IN BAS PROGRAM ***
0486 11D8
0486 11D8 REM *** CLINIC ID ***
0486 11D8 100 CL1.COD$="DDAA" 'DEFAULT CLINIC CODE
048F 11DC
048F 11DC REM ... CLINIC CODE WHEN SINGLE BUBBLED ...
048F 11DC IF MID$(TEXT$,30,1)=" " THEN 102
04A7 11DC X=VAL(MID$(TEXT$,30,1))
048D 11DE CL1.COD$=CLIBUB$(X)
04CF 11DE
04CF 11DE REM *** VISIT DATE ***
04CF 11DE 102 CM$=MID$(DATE$,1,2) 'CURRENT MONTH
04E1 11E2 YR$=MID$(DATE$,9,2) 'CURRENT YEAR
04F3 11E6 X$=MID$(TEXT$,31,4) 'MONTH AND DAY FROM FORM
0505 11EA IF LEFT$(X$,2)<=CM$ THEN 104 'OK, USE THIS YEAR
051A 11EA YR$=RIGHT$(STR$(VAL(YR$)-1),2) 'USE LAST YEAR
053A 11EA 104 VDATE$ = YR$ + X$
0548 11EE 'EDIT VISIT DATE
0548 11EE CK.5000$=VDATE$
0551 11F2 GOSUB 5000 'DATE CHECK
0556 11F2 CK.5010$=VDATE$
055F 11F6 GOSUB 5010 'NUMERIC STRING CHECK
0564 11F6 IF RT.5000 = 0 AND RT.5010 = 0 THEN GOTO 106
0588 11FA N.ERR=N.ERR+1
0590 11FA ED.MSG$(N.ERR)="TODAYS DATE " + DATEERR$(RT.5000)
0586 11FA
0586 11FA REM *** PROVIDER 1 ID (PREFIX + NUM) ***
0586 11FA 106 PROV1.PFX$ = MID$(TEXT$,35,1)
05C8 11FE PROV1.NUM$ = MID$(TEXT$,36,4)
05DA 1202 PR1.COD$=PROV1.PFX$ + PROV1.NUM$
05E8 1206
05E8 1206 SSN.OFFSET = 40
05EF 1208 FMP.OFFSET = 49
05F6 120A RSI.OFFSET = 51
05FD 120C RPF.OFFSET = 53
0604 120E REF.OFFSET = 54
0608 1210 TIM.OFFSET = 57
0612 1212 UPR.OFFSET = 58

```

Offset	Data	Source Line
0619	1214	PRC.OFFSET = 63
0620	1216	SPE.OFFSET = 81
0627	1218	TOT.OFFSET = 50
062E	121A	POINTER = 0
0635	121C	
0635	121C	REM *** REPEAT THE FOLLOWING CODE 4 TIMES ***
0635	121C	FOR I910= 1 TO 4
063C	121C	
063C	121C	REM *** SSN AND FMP ***
063C	121C	110 SSN.COD\$(I910)=MID\$(TEXT\$,SSN.OFFSET+POINTER,9)
065F	121E	FMP.COD\$(I910)=MID\$(TEXT\$,FMP.OFFSET+POINTER,2)
0682	121E	IF SSN.COD\$(I910)+FMP.COD\$(I910)=STRING\$(11," ") THEN 130
06AF	121E	
06AF	121E	REM *** REFERRAL CODE ***
06AF	121E	112 X\$=MID\$(TEXT\$,RSI.OFFSET+POINTER,2) 'SINGLE BUBBLE CODE
06C6	121E	IF X\$=" " THEN 114
06D4	121E	X=VAL(X\$)
06E1	121E	REF.COD\$(I910)=REFCOD\$(X) 'USE TABLE
06FE	121E	GOTO 116
0702	121E	
0702	121E	114 X=VAL(MID\$(TEXT\$,RPF.OFFSET+POINTER,1))
071D	121E	REF.COD\$(I910)=CLINIC2.PFX\$(X)+MID\$(TEXT\$,REF.OFFSET+POINTER,3)
0751	121E	
0751	121E	REM *** TIME SPENT ***
0751	121E	116 X=VAL(MID\$(TEXT\$,TIM.OFFSET+POINTER,1))
076C	121E	PR1.TIM\$(I910)=PROVIDER.TIM\$(X)
0789	121E	IF PR1.TIM\$(I910)<>"000" THEN 118
07A1	121E	
07A1	121E	N.ERR=N.ERR+1
07A9	121E	ED.MSG\$(N.ERR)="NO TIME CODED FOR PART "+STR\$(I910)
07C6	121E	
07C6	121E	REM *** ADDITIONAL PROCEDURE
07C6	121E	118 ADDP\$=MID\$(TEXT\$,UPR.OFFSET+POINTER,5)
07DD	1222	
07DD	1222	REM *** PROCEDURE CODES FOR PROV 1 ***
07DD	1222	120 X\$ = MID\$(TEXT\$,PRC.OFFSET+POINTER,N.PROC)
07F5	1222	GOSUB 6000 'DECODE THE X\$ STRING
07FA	1222	IF TOT = 0 THEN GOTO 122
0809	1224	FOR I910.A= 1 TO TOT
0816	1226	PTR=VAL(HOLD\$(I910.A))
082C	122A	GROUP1\$(I910,I910.A)=PROCD\$(PTR)
084E	122A	NEXT I910.A
085F	122A	
085F	122A	REM ... ADD UNLISTED CODE IF THERE TO PROV 1 ...
085F	122A	122 IF ADDP\$=" " THEN 124
086D	122A	TOT=TOT+1
0875	122A	GROUP1\$(I910,TOT)=ADDP\$
088E	122A	
088E	122A	124 TOT.PROC(I910)=TOT
089D	122A	IF TOT>0 THEN 126
08A8	122A	
08A8	122A	N.ERR=N.ERR+1
08A8	122A	ED.MSG\$(N.ERR)="NO PROCEDURE CODED FOR PART "+STR\$(I910)

RACPB20
EKG (DDAA) FORM

PAGE 9
04 29-87
08:18:35

IBM Personal Computer BASIC Compiler V1.00

```
Offset Data Source Line
08CD 122A
08CD 122A REM *** SPECIFIC PREASSIGNED CL CODES ***
08CD 122A 126 INP.STOS= MID$(TEXT$,SPE.OFFSET+POINTER,9)
08E4 122E GOSUB 5700 'CONVERT ARRAY
08E9 122E SPE.COD$(1910)=BUF.STOS
08FD 1232
08FD 1232 130 POINTER = POINTER + TOT.OFFSET
0908 1232 NEXT 1910
091A 1232 199 REM
091B 1232
091B 1232 REM -----END OF MODULE RACM820.M01-----
091B 1232
091B 1232 IF N.ERR = 0 THEN GOTO 997
092A 1232 LPRINT "LITHO # ";LITHOS;" ... ERRORS"
093C 1232 FOR 1997 = 1 TO N.ERR
0949 1234 LPRINT USING "### ";1997;
0955 1236 LPRINT "==> ";ED.MSG$(1997)
0968 1236 NEXT 1997
097C 1236 LN.COUNT = LN.COUNT + N.ERR + 1
0988 1238 CNTRLOPT = 6
098F 1238 GOSUB 9010 'REJECT THE FORM
0994 1238 GOTO 998 'BYPASS THE DISK WRITER....
0998 1238
0998 1238 997 REM $INCLUDE: 'RACM820.M02' REM INCLUDE THE BASE ENCOUNTER FORM DISK WRITER
0999 1238 REM *****
0999 1238 REM **** AMBULATORY CARE INFORMATION SYSTEM 14 MAR 87 ****
0999 1238 REM **** D R BOLLING ****
0999 1238 REM **** MODULE NAME : RACM820.M02 ****
0999 1238 REM **** SCANNER PROGRAM # : 820- EKG (DDAA) FORM ****
0999 1238 REM ****
0999 1238 REM **** PURPOSE : CREATE AND WRITE THE DISK ****
0999 1238 REM **** RECORD FOR INPUT TO FOCUS ****
0999 1238 REM ****
0999 1238 REM **** PROGRAM ADDS PREFIX TO LITHO FOR EACH PATIENT ****
0999 1238 REM *****
0999 1238 REM **** RESERVED LINE NUMBERS 200-299 ****
0999 1238 REM *****
0999 1238
0999 1238 REM BUILD THE OUTPUT RECORD
0999 1238
0999 1238 GOSUB 270 'BUILD THE RECORD KEY
099E 1238
099E 1238 REM *****
099E 1238 REM **** RECORD TYPE "1" - RECKEY PLUS TYPE 1 FIELDS ****
099E 1238 REM *****
099E 1238 REM RECOUNTS ="8201"+RECKEY$ 'TRANSACTION IDENTIFIER
099E 1238
099E 1238 FOR 1910 = 1 TO 4
09A5 1238 RECOUNTS = "8201" + RECKEY$ + RIGHT$(STR$(1910),1) + RIGHT$(LITHOS,7)
09D5 1240 PTIDS= SSN.COD$(1910) + FMP.COD$(1910)
09F4 1244 IF PTIDS= STRING$(11," ") THEN 210 'NO MORE TO DO
0A09 1244 MID$(RECOUNTS,20,11)=PTIDS
0A18 1244 TYPE1$ ="821"+PR1.TIM$(1910)+STRING$(10," ")+REF.COD$(1910)
```

Offset	Date	Source Line
0A4C	1248	RECOU\$ =RECOU\$ + TYPE1\$ + STRING\$(16," ")
0A67	1248	GOSUB 280
0A6C	1248	PRINT #1,RECOU\$
0A77	1248	NEXT 1910
0A89	1248	
0A89	1248	210 REM ... NO MORE ON THIS FORM ...
0A8A	1248	
0A8A	1248	REM *****
0A8A	1248	REM **** RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE ****
0A8A	1248	REM *****
0A8A	1248	REM RECOU\$ ="8202"+RECKEY\$ 'TRANSACTION IDENTIFIER
0A8A	1248	
0A8A	1248	FOR 1910 = 1 TO 4
0A91	1248	FOR 1910.A = 1 TO TOT.PROC(1910)
0AA5	124A	RECOU\$ = "8202" + RECKEY\$ + RIGHT\$(STR\$(1910),1) + RIGHT\$(LITHOS,7)
0AD5	124A	PTID\$= SSN.COD\$(1910) + FMP.COD\$(1910)
0AF4	124A	IF PTID\$= STRING\$(11," ") THEN 250 'NO MORE TO DO
0B09	124A	MID\$(RECOU\$,20,11)=PTID\$
0B18	124A	RECOU\$ =RECOU\$ + "821" + GROUP1\$(1910,1910.A)
0B3C	124A	GOSUB 280
0B41	124A	PRINT #1,RECOU\$
0B4C	124A	NEXT 1910.A
0B60	124A	250 NEXT 1910
0B72	124A	
0B72	124A	REM *****
0B72	124A	REM **** RECORD TYPE "3" - RECKEY PLUS SPEC PRE CLINIC CODES ****
0B72	124A	REM *****
0B72	124A	
0B72	124A	FOR 1910 = 1 TO 4
0B79	124A	IF LEN(SPE.COD\$(1910))=0 THEN 260
0B8F	124A	RPOINT = 1
0B96	124C	
0B96	124C	252 RECOO3\$=RIGHT\$(MID\$(SPE.COD\$(1910),RPOINT,2),1)
0B88	1250	IF RECOO3\$=" " THEN 254
0BC6	1250	RECOU\$ = "8203" + RECKEY\$ + RIGHT\$(STR\$(1910),1) + RIGHT\$(LITHOS,7)
0BF6	1250	PTID\$= SSN.COD\$(1910) + FMP.COD\$(1910)
0C15	1250	IF PTID\$= STRING\$(11," ") THEN 260 'NO MORE TO DO
0C2A	1250	MID\$(RECOU\$,20,11)=PTID\$
0C39	1250	RECOU\$ =RECOU\$ + "82" + RECOO3\$
0C4C	1250	GOSUB 280
0C51	1250	PRINT #1,RECOU\$
0C5C	1250	
0C5C	1250	254 RPOINT = RPOINT + 2
0C65	1250	IF RPOINT < LEN(SPE.COD\$(1910)) THEN 252
0C70	1250	
0C70	1250	260 NEXT 1910
0C8F	1250	
0C8F	1250	REM END OF TYPE 3 RECORDS
0C3F	1250	
0C8F	1250	GOTO 299
0C93	1250	
0C93	1250	REM *****
0C93	1250	REM **** SUBROUTINE 270 - BUILD THE RECORD KEY ****
0C93	1250	REM *****

Offset	Data	Source Line
0C93	1250	270 RECKEY\$=""
0C9C	1250	
0C9C	1250	REM *** CLINIC ID (PREFIX + CODE) ***
0C9C	1250	RECKEY\$= CL1.COD\$
0CA5	1250	
0CA5	1250	REM *** VISIT DATE ***
0CA5	1250	RECKEY\$=RECKEY\$+ VDATES
0CB1	1250	
0CB1	1250	REM *** PROVIDER ID (PREFIX + NUM) ***
0CB1	1250	RECKEY\$ = RECKEY\$ + PROV1.PFX\$ + PROV1.NUM\$
0CC4	1250	
0CC4	1250	REM *** PATIENT ID (SSN+FMP) ***
0CC4	1250	RECKEY\$ = RECKEY\$ + STRING\$(11," ")
0CD6	1250	
0CD6	1250	REM *** LITHO CODE ***
0CD6	1250	REM DO LITHO IN LOOP ABOVE
0CD6	1250	REM RECKEY\$ = RECKEY\$ + LITHOS
0CD6	1250	
0CD6	1250	RETURN
0CD9	1250	
0CD9	1250	REM *****
0CD9	1250	REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
0CD9	1250	REM *****
0CD9	1250	280 PAD=MAXLENGTH - LEN(RECOUT\$) 'FIND OUT HOW SHORT THE RECORD IS
0CE7	1252	RECOUT\$ = RECOUT\$ + STRING\$(PAD,PAD\$) 'PAD THE RECORD WITH FILL CHAR
0CFA	1252	RETURN
0CFD	1252	
0CFD	1252	
0CFD	1252	299 REM
0CFE	1252	
0CFE	1252	REM -----END OF MODULE RXXM820.M02-----
0CFE	1252	
0CFE	1252	998 REM CONTINUE
0CFF	1252	
0CFF	1252	999 READTYPE = 2
0D06	1252	IF LN.COUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
0D16	1252	GOTO 10
0D1A	1252	
0D1A	1252	REM END OF SCAN/DECODE/WRITE LOOP =====
0D1A	1252	
0D1A	1252	1000 REM \$INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
0D1B	1252	REM *****
0D1B	1252	REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
0D1B	1252	REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
0D1B	1252	REM *****
0D1B	1252	REM PATIENT OMR INPUT PROGRAM *
0D1B	1252	REM *
0D1B	1252	REM This program verifies user is logged on properly. If there is no *
0D1B	1252	REM valid user logged on at the time of execution, this subroutine will*
0D1B	1252	REM chain to the logon program RACP05, otherwise a return is issued. *
0D1B	1252	REM *****
0D1B	1252	REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
0D1B	1252	REM *****
0D1B	1252	1001 OPEN ":",1,"RACLOG.DAT"

Offset	Data	Source Line
002D	1252	IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
003B	1252	INPUT #1,USERS(1),DTS,TMS,PIDS
005C	125E	IF USERS(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
006A	125E	IF USERS(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
0078	125E	CLOSE 1
007F	125E	SCREEN 0,1,0,0
0095	125E	COLOR FORE,BACK,BORD
00AB	125E	CLS
00AF	125E	RETURN
00B2	125E	1002 CLOSE
00B6	125E	CHAIN "RACP05"
00BD	125E	'=====END OF LOGON VERIFY SUBROUTINE 1000=====
00BD	125E	2000 REM \$INCLUDE: 'RACS2000.SUB' INCLUDE THE REPLY/DELAY SUB
00BE	125E	REM *****
00BE	125E	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
00BE	125E	REM **** SKIP COLE ****
00BE	125E	REM **** SUBROUTINE NAME : RACS2000.SUB ****
00BE	125E	REM **** SCANNER PROGRAM # : ALL ****
00BE	125E	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
00BE	125E	REM **** SERVERS AS A WAIT AND REPLY ****
00BE	125E	REM **** ENTRY MODULE ****
00BE	125E	REM **** INPUT : SINGLE KEYBOARD ENTRY ****
00BE	125E	REM **** ****
00BE	125E	REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
00BE	125E	REM **** ****
00BE	125E	REM **** RESERVED LINE ****
00BE	125E	REM **** NUMBERS : 2001-2010 ****
00BE	125E	REM *****
00BE	125E	2001 REM REPLY FUNCTION
00BF	125E	2002 REPLY\$=INKEY\$: IF REPLY\$="" THEN 2002
00D3	1262	REPLY=ASC(REPLY\$)
00DD	1264	IF REPLY > 90 THEN REPLY\$=CHR\$(REPLY XOR 32) 'CONVERT TO CAPS
00FB	1264	IF REPLY\$ < "A" OR REPLY\$ > "Z" THEN REPLY\$="?"
0E24	1264	RETURN
0E27	1264	0E27 1264 5000 REM \$INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB
0E28	1264	REM *****
0E28	1264	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0E28	1264	REM **** SKIP COLE ****
0E28	1264	REM **** SUBROUTINE NAME : RXXS5000.SUB ****
0E28	1264	REM **** SCANNER PROGRAM # : ALL ****
0E28	1264	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0E28	1264	REM **** PERFORMS A DATE EDIT ****
0E28	1264	REM **** ****
0F28	1264	REM **** INPUT : DATE TO BE CHECKED MUST BE ****
0E28	1264	REM **** IN THE VARIABLE NAMED ****
0E28	1264	REM **** 'CK.5000\$' ****
0E28	1264	REM **** IN THE FORMAT "YYMMDD" ****
0E28	1264	REM **** ****
0E28	1264	REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
0E28	1264	REM **** VARIABLE. IF THIS VARIABLE ****
0E28	1264	REM **** CONTAINS ANY NUMBER OTHER ****

Offset Data Source Line IBM Personal Computer BASIC Compiler V*.00

```

0E28 1264 REM ****          THAN 0, AN ERROR WAS FOUND ****
0E28 1264 REM ****          IN THE DATE.          ****
0E28 1264 REM ****          ****
0E28 1264 REM ****    RESERVED LINE          ****
0E28 1264 REM ****          NUMBERS      : 5001-5009          ****
0E28 1264 REM ****          ****
0E28 1264          RT.5000 = 0
0E2F 1264          CKYEAR = VAL(LEFT$(CK.5000$,2))    'YEAR NUMERIC VALUE
0E42 1266          CKMONTH = VAL(MID$(CK.5000$,3,2))    'MONTH NUMERIC VALUE
0E58 1268          CKDAY = VAL(RIGHT$(CK.5000$,2))    'DAY NUMERIC VALUE
0E68 126A
0E68 126A          IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
0E81 126A          IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
0E97 126A          IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
0EAD 126A          IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
0EC3 126A
0EC3 126A REM    LEAP YEAR CHECK
0EC3 126A          MOLENGTH(2) = 28
0ECA 126A          IF CKMONTH<> 2          THEN GOTO 5005 'MUST BE FEBRUARY
0ED9 126A          IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR
0EEE 126A          MOLENGTH(2) = 29
0EF5 126A
0EF5 126A 5005  IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
0F14 126A
0F14 126A 5009  RETURN
0F17 126A
0F17 126A REM -----END OF SUBROUTINE 5000 -----
0F17 126A
0F17 126A 5010  REM $INCLUDE: 'RACS5010.SUB'  INCLUDE THE NUMERIC STRING EDITOR
0F18 126A REM *****
0F18 126A REM ****    AMBULATORY CARE DATA BASE          1 MAY 85    ****
0F18 126A REM ****                                SKIP COLE    ****
0F18 126A REM ****    SUBROUTINE NAME      :  RXXS5010.SUB    ****
0F18 126A REM ****    SCANNER PROGRAM #    :  ALL            ****
0F18 126A REM ****    FUNCTION              :  THIS SUBROUTINE MODULE    ****
0F18 126A REM ****                                PERFORMS A NUMERIC STRING    ****
0F18 126A REM ****                                EDIT.            ****
0F18 126A REM ****                                ****
0F18 126A REM ****    INPUT                  :  STRING TO BE EDITED IS IN    ****
0F18 126A REM ****                                THE VARIABLE NAMED    ****
0F18 126A REM ****                                'CK.5010$'    ****
0F18 126A REM ****                                ****
0F18 126A REM ****    OUTPUT                  :  'RT.5010' IS THE RETURN CODE    ****
0F18 126A REM ****                                VARIABLE. IF THIS VARIABLE    ****
0F18 126A REM ****                                CONTAINS ANY NUMBER OTHER    ****
0F18 126A REM ****                                THAN 0, AN ERROR WAS FOUND    ****
0F18 126A REM ****                                IN THE STRING.    ****
0F18 126A REM ****                                ****
0F18 126A REM ****    RESERVED LINE          ****
0F18 126A REM ****          NUMBERS      :  5011-5019          ****
0F18 126A REM ****          ****
0F18 126A          RT.5010 = 0
0F1F 126A
0F1F 126A          FOR I5010 = 1 TO LEN(CK.5010$)

```

Offset	Data	Source Line
0F2F	126C	J5010= ASC(MID\$(CK.5010\$,15010,1))
0F43	1270	IF J5010 < 48 OR J5010 > 57 THEN RT.5010 = RT.5010 + 1
0F6B	1270	NEXT 15010
0F7C	1270	RETURN
0F7F	1270	REM ----- END OF SUBROUTINE 5010 -----
0F7F	1270	5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTPATIENT UCA CHECK SUB
0F80	1270	5700 REM \$INCLUDE: 'RACS5700.SUB' INCLUDE THE ARRAY CONVERTER SUB
0F81	1270	*****
0F81	1270	REM **** AMBULATORY CARE DATA BASE 29 JUL 85 ****
0F81	1270	REM **** D R BOLLING ****
0F81	1270	REM **** SUBROUTINE NAME : RXXS5700.SUB ****
0F81	1270	REM **** SCANNER PROGRAM # : ALL ****
0F81	1270	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0F81	1270	CONVERTS A BINARY ARRAY INTO ****
0F81	1270	TWO CHAR CODES. ****
0F81	1270	REM ****
0F81	1270	REM **** INPUT : INP.STO\$ AS STRING ****
0F81	1270	REM ****
0F81	1270	REM ****
0F81	1270	REM **** OUTPUT : BUF.STO\$ AS STRING. ****
0F81	1270	REM ****
0F81	1270	REM ****
0F81	1270	REM **** RESERVED LINE ****
0F81	1270	REM **** NUMBERS : 5710-5730 ****
0F81	1270	*****
0F81	1270	BUF.STO\$=""
0F8A	1270	N.STO=1
0F91	1272	5710 X.STO=INSTR(N.STO,INP.STO\$,"1")
0FA3	1274	IF X.STO=0 THEN GOTO 5720 'THATS ALL
0FB2	1274	N.STO = X.STO + 1 'NEXT STARTING POINT
0FBA	1274	X.STO = X.STO + 100 'PAD WITH LEADING ZERO
0FC4	1274	BUF.STO\$ = BUF.STO\$ + RIGHT\$(STR\$(X.STO),2)
0FDA	1274	IF N.STO <= LEN(INP.STO\$) THEN GOTO 5710
0FED	1274	5720 RETURN
0FF0	1274	REM ----- END OF SUBROUTINE 5700 -----
0FF0	1274	6000 REM \$INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE ROUTINE
0FF1	1274	*****
0FF1	1274	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0FF1	1274	REM **** SKJP COLE ****
0FF1	1274	REM **** SUBROUTINE NAME : RXXS6000.SUB ****
0FF1	1274	REM **** SCANNER PROGRAM # : ALL ****
0FF1	1274	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0FF1	1274	PERFORMS INSTRING SEARCH ****
0FF1	1274	REM ****
0FF1	1274	REM **** INPUT : STRING TO BE SEARCHED MUST ****
0FF1	1274	BE IN THE VARIABLE NAMED : ****
0FF1	1274	REM **** 'X\$' ****
0FF1	1274	REM ****
0FF1	1274	REM **** OUTPUT : 'TOT' = TOTAL NUMBER OF ****

Offset	Data	Source Line
OFF1	1274	REM **** HITS IN THE DESTING ****
OFF1	1274	REM **** 'HOLD\$()' IS THE ARRAY ****
OFF1	1274	REM **** CONTAINING THE NUMERIC ****
OFF1	1274	REM **** VALUE OF THE HIT POSITIONS ****
OFF1	1274	REM ****
OFF1	1274	REM *****
OFF1	1274	REM **** RESERVED LINE ****
OFF1	1274	REM **** NUMBERS : 6001-6009 ****
OFF1	1274	REM *****
OFF1	1274	6001 PTR = INSTR(X\$, "1")
OFFF	1274	TOT = 0
1006	1274	WHILE PTR > 0
1011	1274	TOT=TOT+1
1019	1274	HOLD\$(TOT) = RIGHT\$(STR\$(PTR), 2)
1038	1274	PTR=PTR+1
1043	1274	PTR = INSTR(PTR, X\$, "1")
1055	1274	WEND
1059	1274	RETURN
105C	1274	
105C	1274	REM -----END OF SUBROUTINE RXXS6000.SUB-----
105C	1274	
105C	1274	
105C	1274	7000 REM \$INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
105D	1274	REM *****
105D	1274	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
105D	1274	REM **** SKIP COLE ****
105D	1274	REM **** SUBROUTINE NAME : RACS7000.SUB ****
105D	1274	REM **** SCANNER PROGRAM # : ALL ****
105D	1274	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
105D	1274	REM **** PRINTS THE STANDARD SCREEN ****
105D	1274	REM **** HEADING. ****
105D	1274	REM **** INPUT : COMMON VARIABLE USERS\$(2) ****
105D	1274	REM **** SYSTEM DATE ****
105D	1274	REM ****
105D	1274	REM **** OUTPUT : SCREEN HEADING ****
105D	1274	REM ****
105D	1274	REM **** RESERVED LINE ****
105D	1274	REM **** NUMBERS : 7001-7010 ****
105D	1274	REM *****
105D	1274	
105D	1274	7001 LOCATE 1,1
1067	1274	PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
106F	1274	LOCATE 1,65
107C	1274	PRINT DATES;
1084	1274	LOCATE 2,1
1091	1274	PRINT "USER : ";USERS\$(1)
109E	1274	RETURN
10A1	1274	7100 REM \$INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
10A2	1274	REM *****
10A2	1274	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
10A2	1274	REM **** SKIP COLE ****
10A2	1274	REM **** SUBROUTINE NAME : RXXS7100.SUB ****
10A2	1274	REM **** SCANNER PROGRAM # : ALL ****
10A2	1274	REM **** FUNCTION : THIS SUBROUTINE MODULE ****

Offset	Data	Source Line
10A2	1274	REM **** PRINTS THE STANDARD HEADING ****
10A2	1274	REM **** ON THE PRINTER. ****
10A2	1274	REM **** INPUT : DATE,PAGE,PGMID\$,PGMTITL\$ ****
10A2	1274	REM **** OUTPUT : PRINTER HEADING, LN.COUNT ****
10A2	1274	REM **** RESERVED LINE ****
10A2	1274	REM **** NUMBERS : 7101-7110 ****
10A2	1274	REM *****
10A2	1274	7101 IF PAGE > 0 THEN LPRINT CHR\$(12);
10B8	1274	LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITL\$;
10C5	1274	LPRINT TAB(70);DATE\$
10D8	1274	PAGE=PAGE+1
10E0	1274	LPRINT "PROGRAM ";PGMID\$;TAB(70);"PAGE";
10FD	1274	LPRINT USING "####";PAGE
1109	1274	LPRINT
1111	1274	LN.COUNT=3
1118	1274	RETURN
1118	1274	
111B	1274	8000 REM \$INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
111C	1274	REM *****
111C	1274	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ***
111C	1274	REM **** SKIP COLE ***
111C	1274	REM **** SUBROUTINE NAME : RXXS8000.SUB ***
111C	1274	REM **** SCANNER PROGRAM # : ALL ***
111C	1274	REM **** FUNCTION : THIS SUBROUTINE MODULE ***
111C	1274	REM **** IS A GROUPING THAT PERFORMS ***
111C	1274	REM **** VARIOUS DECODING FUNCTIONS ***
111C	1274	REM **** ON THE SCANNER DATA ***
111C	1274	REM ****
111C	1274	REM **** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
111C	1274	REM **** 8050 - CHECK FOR END OF JOB ***
111C	1274	REM **** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
111C	1274	REM **** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
111C	1274	REM **** (RETURNED IN TEXT\$ STRING VARIABLE) ***
111C	1274	REM ****
111C	1274	REM **** INPUT : SHEET RECORD, RECORD LENGTH ***
111C	1274	REM ****
111C	1274	REM **** OUTPUT : 'TEXT\$' TRING VARIABLE ***
111C	1274	REM ****
111C	1274	REM **** RESERVED LINE ***
111C	1274	REM **** NUMBERS : 8001-8500 ***
111C	1274	REM *****
111C	1274	
111C	1274	'DECODE THE HEADER ONLY
111C	1274	8001 POINTER = 0
1123	1274	RECORDPTR = VARPTR(SHEETREC(0))
112A	1276	FOR J8000 = 1 TO 21
1131	1276	8002 TEXT\$= TEXT\$+CHR\$(PEEK(RECORDPTR + POINTER))
114F	1276	POINTER=POINTER+1
1157	1276	NEXT J8000
1156	1278	PROGRAM\$= LEFT\$(TEXT\$,3)
1175	127C	BATCH\$= MID\$(TEXT\$,4,3)

Offset	Data	Source Line
1187	1280	SERIAL\$= MID\$(TEXT\$,7,4)
1199	1284	RUNID\$= MID\$(TEXT\$,11,1)
11AB	1288	FORM\$= MID\$(TEXT\$,12,2)
11BD	128C	POCKET\$= MID\$(TEXT\$,14,1)
11CF	1290	SCANERR1\$=MID\$(TEXT\$,16,2)
11E1	1294	SCANERR2\$=MID\$(TEXT\$,18,2)
11F3	1298	SCANERR3\$=MID\$(TEXT\$,20,2)
1205	129C	GOTO 8500
1209	129C	
1209	129C	8050 REM CHECK FOR END OF JOB/END OF BATCH
120A	129C	IF PROGRAM\$ = PGMID\$ THEN GOTO 8500
121C	129C	LPRINT STRING\$(80,"*")
122A	129C	LPRINT
1232	129C	LPRINT "RECORDS PROCESSED ... ";SERIAL\$
123F	129C	LPRINT "STARTED AT ";BTIM\$
124C	129C	LPRINT "ENDED AT ";TIM\$
1259	129C	LPRINT CHR\$(12)
1264	129C	GOTO 30000
1268	129C	
1268	129C	8070 REM CHECK FOR SCANNER ERRORS
1269	129C	IF POCKET\$ = " " GOTO 8500
127B	129C	LPRINT LITHOS;
1283	129C	LPRINT " ... SCANNER ERRORS : ";
1288	129C	LPRINT SCANERR1\$;" / ";
1298	129C	LPRINT SCANERR2\$;" / ";
12A5	129C	LPRINT SCANERR3\$
12AD	129C	LN=LN+1
12B5	129E	GOTO 999
12B9	129E	
12B9	129E	8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
12BA	129E	LOCATE 5,1:PRINT "PROGRAM ";PROGRAM\$;
12D4	129E	
12D4	129E	PRINT " BATCH ";BATCH\$;
12E1	129E	PRINT " RUN ";RUNID\$;
12EE	129E	PRINT " FORM ";FORM\$;
12FB	129E	PRINT " POCKET ";POCKET\$
1308	129E	GOTO 8500
130C	129E	
130C	129E	8200 REM DECODE THE RESPONSE POSITIONS
130D	129E	POINTER = 21
1314	129E	RECORDPTR = VARPTR(SHEETREC(0))
131B	129E	FOR J8000 = 22 TO RECORDLENGTH
1328	12A0	8202 TEXT\$ = TEXT\$+CHR\$(PEEK(RECORDPTR + POINTER))
1346	12A0	POINTER=POINTER+1
134E	12A0	NEXT J8000
135F	12A0	
135F	12A0	8500 RETURN
1362	12A0	
1362	12A0	REM ----- END OF RXXS8000.SUB -----
1362	12A0	
1362	12A0	9000 REM \$INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
1363	12A0	REM *****
1363	12A0	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1363	12A0	REM **** SKIP COLE ****

Offset	Data	Source Line
1363	12A0	REM **** PROGRAM NAME : RACS9000.SUB ****
1363	12A0	REM **** SCANNER PROGRAM # : ALL ****
1363	12A0	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1363	12A0	REM **** CONTROLS THE SCANNER I/O ****
1363	12A0	REM ****
1363	12A0	REM **** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS ****
1363	12A0	REM **** COMMUNICATIONS MANUAL AND THE ****
1363	12A0	REM **** PRE-RELEASED SOFTWARE GUIDE ****
1363	12A0	REM ****
1363	12A0	REM *****
1363	12A0	REM **** RESERVED LINE ****
1363	12A0	REM **** NUMBERS : 9001-9100 ****
1363	12A0	REM *****
1363	12A0	REM *****
1363	12A0	REM *****
1363	12A0	REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
1363	12A0	REM **** ARGUMENTS: PRESET ... SEE BELOW ****
1363	12A0	REM *****
1363	12A0	9001 REM
1364	12A0	PROTOCOL(0) = 9600 'BAUD RATE
1368	12A0	PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
1372	12A0	PROTOCOL(2) = 8 'DATA BITS
1379	12A0	PROTOCOL(3) = 1 'STOP BITS
1380	12A0	PROTOCOL(4) = 2 'RS-232 PORT
1387	12A0	PROTOCOL(5) = 0 'WRITE TIME-OUT
138E	12A0	PROTOCOL(6) = 0 'READ TIME-OUT
1395	12A0	
1395	12A0	ERRSTAT\$ = SPACE\$(60)
13A1	12A0	ARGPTR = VARPTR(PROTOCOL(0))
13A8	12A2	CALL SETUP (ARGPTR,ERRSTAT\$)
13B9	12A2	ERRMSG\$=""
13C2	12A2	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="SETUP ERROR "+ERRSTAT\$
13DE	12A2	GOTO 9100
13E2	12A2	
13E2	12A2	REM *****
13E2	12A2	REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
13E2	12A2	REM **** ARGUMENTS: CNTRLOPT ****
13E2	12A2	REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
13E2	12A2	REM **** CNTRLOPT = 2 = STOP SCANNER (S0) ****
13E2	12A2	REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
13E2	12A2	REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
13E2	12A2	REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
13E2	12A2	REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
13E2	12A2	REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
13E2	12A2	REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
13E2	12A2	REM *****
13E2	12A2	9010 REM
13E3	12A2	ERRSTAT\$ = SPACE\$(60)
13FF	12A2	CALL CNTRL (CNTRLOPT,ERRSTAT\$)
1400	12A2	ERRMSG\$=""
1409	12A2	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
1425	12A2	GOTO 9100
1429	12A2	

Offset	Data	Source Line
1429	12A2	REM *****
1429	12A2	REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
1429	12A2	REM ****
1429	12A2	REM **** ARGUMENTS: READTYPE ****
1429	12A2	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
1429	12A2	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
1429	12A2	REM ****
1429	12A2	REM **** ARGUMENTS: RECORDLENGTH ****
1429	12A2	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
1429	12A2	REM **** TRANSMITTED ****
1429	12A2	REM *****
1429	12A2	9020 REM
142A	12A2	ERRSTAT\$ = SPACES(60)
1436	12A2	RECORDPTR = VARPTR(SHEETREC(0))
1430	12A2	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
1456	12A2	ERRMSG\$=""
145F	12A2	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
1480	12A2	GOTO 9100
1484	12A2	
1484	12A2	REM *****
1484	12A2	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
1484	12A2	REM ****
1484	12A2	REM **** ARGUMENTS: PRINTPOS ****
1484	12A2	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
1484	12A2	REM **** VALUES = 0 THRU 90 ****
1484	12A2	REM ****
1484	12A2	REM **** ARGUMENTS: PSTRINGS ****
1484	12A2	REM **** TEXT TO BE PRINTED ON THE FORM ****
1484	12A2	REM ****
1484	12A2	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
1484	12A2	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
1484	12A2	REM *****
1484	12A2	9030 REM
1485	12A2	ERRSTAT\$ = SPACES(60)
1491	12A2	RECORDPTR = VARPTR(SHEETREC(0))
1498	12A2	CALL TPRINT(PRINTPOS,PSTRINGS,ERRSTAT\$)
14AD	12A8	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
14C9	12A8	GOTO 9100
14CD	12A8	
14CD	12A8	9100 RETURN
14D0	12A8	REM -----END OF SUBROUTINE RACS9000.SUB -----
14D0	12A8	
14D0	12A8	REM END OF SUBROUTINES =====
14D0	12A8	
14D0	12A8	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
14D1	12A8	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
14D9	12A8	LPRINT "ERASING FILE ";DATFIL\$
14E6	12A8	BEEP
14EA	12A8	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
14F6	12A8	CLOSE
14FA	12A8	OPEN DATFIL\$ FOR OUTPUT AS #1
150C	12A8	PRINT #1,STRING\$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
151E	12A8	CLOSE
1522	12A8	

RACP820
EKG (D0AA) FORM

PAGE 20

04-29-87

08:18:35

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

1522	12A8	30000	REM
1523	12A8		CLOSE
1527	12A8		CHAIN "RACP10"
152E	12A8		END
1532	12A8		
1535	12A8		

22151 Bytes Available

16573 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

PAGE 1

07-06-87

14:57:17

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

001A 0002 REM \$LINESIZE: 132

001A 0002 REM \$PAGESIZE: 66

001A 0002 REM \$TITLE: 'RACP830 '

001A 0002 REM \$SUBTITLE: 'EMERGENCY ROOM FORM DESTING/DECODE PROGRAM'

001A 0002 REM \$PAGE

RACP830

EMERGENCY ROOM FORM DESTROY/DECODE PROGRAM

PAGE 2

07-06-87

14:57:17

IBM Personal Computer BASIC Compiler V1.00

```

Offset Data Source Line
001A 0002 REM +-----+
001A 0002 REM | NAME: RACP830          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | DATE: 11 APR 87      EMERGENCY ROOM FORM PROGRAM      |
001A 0002 REM | D R BOLLING                          |
001A 0002 REM +-----+
001A 0002 REM          EMERGENCY ROOM FORM OMR INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the base form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RACP05, otherwise,
001A 0002 REM the program chains to program RACP10 on exit.
001A 0002 REM
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 REM *****
001A 0002 REM * NAME: RACDIM.MOD          DIMENSION DEFINITIONS *
001A 0002 REM * Date: 28 Feb 84          Written by: Floyd Cole *
001A 0002 REM *****
001A 0002 REM ' Dimensioned variables are defined in this file.
001A 0002 REM ' It is an included file so it cannot be run in a stand-alone,
001A 0002 REM ' mode.
001A 0002 REM '
001A 0002 REM ' This program segment may be modified, but all files containing
001A 0002 REM ' an include for this segment must be re-compiled in order to
001A 0002 REM ' affect the changes made here.
001A 0002 REM ' ***** START OF DIMENSION DEFINITION *****
001A 0002 REM
001A 0002 REM          DEFINIT A-Z
001A 0002 REM          DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002 REM
001A 0002 REM ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002 REM
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002 REM
001A 0002 REM          DIM SHEETREC(1750)      '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 REM          DIM PROTOCOL(7)         '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 REM          DIM YN$(3)              '(YES/NO ANSWERS 0=?, 1 = Y , 2=N , 3=X)
001A 0002 REM          DIM ED.MSG$(30)         '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002 REM          DIM CLINIC1.PFX$(5)     '(PREFIX -B D F G S- FOR CLINIC #1)
001A 0002 REM          DIM CLINIC2.PFX$(6)     '(PREFIX -A B C D F S- FOR CLINIC #2)
001A 0002 REM          DIM PROVIDER.TIMES(22)  '(TIME TABLE FOR PROVIDERS)
001A 0002 REM          DIM PROCED$(125)        '(PROCEDURE TABLE FOR BASE FORM)
001A 0002 REM          DIM DIAGN.TAB$(225)     '(DIAGNOSIS TABLE FOR BASE FORM)

```

```

Offset  Date   Source Line
001A 0002      DIM HOLDS(99)      '(HOLD AREA FOR SUBROUTINE 6000)
001A 0002      DIM SPECLS(09)    '(SPECIAL PROGRAMS)
001A 0002
001A 0002  REM $INCLUDE: 'RACCMN.MOD'  REM INCLUDE THE COMMON AREA DEFINITION
001A 0002  *****
001A 0002  '*  NAME: RACCMN.MOD              COMMON AREA DEFINITION  *
001A 0002  '*  Date: 28 Feb 84                Written by: Floyd Cole  *
001A 0002  *****
001A 0002  '  COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE.  IT IS AN
001A 0002  '  INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE,  MODE.
001A 0002  '
001A 0002  '  This program segment may be modified, but all files containing
001A 0002  '  an include for this segment must be re*compiled in order to
001A 0002  '  affect the changes made here.
001A 0002  '
001A 0002  '  *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002      COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELLS 'BASIC SCREEN COLORS
001A 0002      COMMON HEADERS      '21 CHARACTER SCANNER HEADER INFO
001A 0002      COMMON TEXT$      ' ' AINING CHARACTERS FROM SCANNER
001A 0002      COMMON PGMIDS      'PROGRAM OR FORM ID
001A 0002      COMMON MOLENGTH()      'DAYS IN THE MONTH
001A 0002      COMMON USERS()
001A 0002  '  *****END  OF COMMON DEFINITION*****
001A 0002
001A 0002
001A 0002  REM $INCLUDE: 'RACDEF.MOD'  REM INCLUDE THE DEFAULT DEFINITIONS
001A 0002  *****
001A 0002  '*  NAME: RACP01.DEF              DEFAULT DEFINITIONS  *
001A 0002  '*  Date: 28 Feb 84                Written by: Floyd Cole  *
001A 0002  *****
001A 0002  '  Variables used in common that have a default value on start*up
001A 0002  '  will be held in this file.  It is an included file so it cannot
001A 0002  '  be run in a stand*alone mode.  In normal operation, this file
001A 0002  '  should be 'included' in the main program only (RACP10.BAS).
001A 0002  '
001A 0002  '  This program segment may be modified, but all files containing
001A 0002  '  an include for this segment must be re*compiled in order to
001A 0002  '  affect the changes made here.
001A 0002  '
001A 0002  '  *****START OF DEFAULT DEFINITION*****
001A 0002
001A 0002      FORE = 15      'FOREGROUND COLOR = INTENSE WHITE
0048 1624      BACK = 1      'Background Color = Light Blue
004F 1624      BORD = 4      'BORDER              = RED
0056 1626      HIDE = 4      'ALTERNATE COLOR = RED
005D 1626      EFORE = 14      'ERROR FOREGROUND DISPLAY
0064 1626      EBACK = 0      'ERROR BACKGROUND DISPLAY
0068 1626      BELLS = CHR$(7) 'Sound the bell
0077 1626
0077 1626      MOLENGTH(1) = 31      'JAN
007E 1626      MOLENGTH(2) = 28      'FEB <--MODIFIED IN SUBROUTINE RACSS000.SUB
0085 1626      MOLENGTH(3) = 31      'MAR
008C 1626      MOLENGTH(4) = 30      'APR

```

Offset Data Source Line

```

0093 1626      MOLENGTH(5) = 31      'MAY
009A 1626      MOLENGTH(6) = 30      'JUN
00A1 1626      MOLENGTH(7) = 31      'JUL
00A8 1626      MOLENGTH(8) = 31      'AUG
00AF 1626      MOLENGTH(9) = 30      'SEP
00B6 1626      MOLENGTH(10) = 31     'OCT
00BD 1626      MOLENGTH(11) = 30     'NOV
00C4 1626      MOLENGTH(12) = 31     'DEC
00CB 1626
00CB 1626      DATEERR$(0) = " "
00D4 1626      DATEERR$(1) = "INVALID MONTH"
00DD 1626      DATEERR$(2) = "INVALID DAY "
00E6 1626      DATEERR$(3) = "DAY TOO LARGE FOR MONTH CODED"
00EF 1626
00EF 1626      MAXLENGTH = 80        'MAXIMUM LENGTH OF OUTPUT RECORD
00F6 1626      PADS = " "          'PAD CHARACTER FOR SHORT RECORDS
00FF 162C
00FF 162C      ' *****END OF DEFAULT DEFINITION*****
00FF 162C
00FF 162C      KEY OFF
0105 162C
0105 162C      REM *****
0105 162C      REM PROGRAM PARAMETERS
0105 162C
0105 162C      REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0105 162C      REM      CLINIC #1          CLINIC #2
0105 162C      CLINIC1.PFX$(0)=" "      : CLINIC2.PFX$(0)=" "
0117 162C      CLINIC1.PFX$(1)="B"      : CLINIC2.PFX$(1)="A"
0129 162C      CLINIC1.PFX$(2)="D"      : CLINIC2.PFX$(2)="B"
0138 162C      CLINIC1.PFX$(3)="F"      : CLINIC2.PFX$(3)="C"
014D 162C      CLINIC1.PFX$(4)="G"      : CLINIC2.PFX$(4)="D"
015F 162C      CLINIC1.PFX$(5)="S"      : CLINIC2.PFX$(5)="F"
0171 162C      CLINIC2.PFX$(6)="S"
017A 162C
017A 162C      REM *** ENCOUNTER FORM PROVIDER TIME TABLE ***
017A 162C
017A 162C      PROVIDER.TIMES$(0)="000"      ' NO TIME
0183 162C      PROVIDER.TIMES$(1)="002"      ' 2 MINUTES
018C 162C      PROVIDER.TIMES$(2)="005"      ' 5 MINUTES
0195 162C      PROVIDER.TIMES$(3)="010"      ' 10 MINUTES
019E 162C      PROVIDER.TIMES$(4)="015"      ' 15 MINUTES
01A7 162C      PROVIDER.TIMES$(5)="020"      ' 20 MINUTES
01B0 162C      PROVIDER.TIMES$(6)="030"      ' 30 MINUTES
01B9 162C      PROVIDER.TIMES$(7)="045"      ' 45 MINUTES
01C2 162C      PROVIDER.TIMES$(8)="060"      ' 1 HOUR
01CB 162C      PROVIDER.TIMES$(9)="090"      ' 1 HOURS/30 MINUTES
01D4 162C      PROVIDER.TIMES$(10)="120"      ' 2 HOURS
01DD 162C      PROVIDER.TIMES$(11)="150"      ' 2 HOURS/30 MINUTES
01E6 162C      PROVIDER.TIMES$(12)="180"      ' 3 HOURS
01EF 162C      PROVIDER.TIMES$(13)="210"      ' 3 HOURS/30 MINUTES
01F8 162C      PROVIDER.TIMES$(14)="240"      ' 4 HOURS
0201 162C      PROVIDER.TIMES$(15)="270"      ' 4 HOURS/30 MINUTES
020A 162C      PROVIDER.TIMES$(16)="300"      ' 5 HOURS

```

Offset	Data	Source Line
0213	162C	PROVIDER.TIMES(17)="330" ' 5 HOURS/30 MINUTES
021C	162C	PROVIDER.TIMES(18)="360" ' 6 HOURS
0225	162C	PROVIDER.TIMES(19)="390" ' 6 HOURS/30 MINUTES
022E	162C	PROVIDER.TIMES(20)="420" ' 7 HOURS
0237	162C	PROVIDER.TIMES(21)="450" ' 7 HOURS/30 MINUTES
0240	162C	PROVIDER.TIMES(22)="480" ' 8 HOURS
0249	162C	
0249	162C	REM YES/NO TABLE
0249	162C	YNS(0)=" " : YNS(1)="Y" : YNS(2)="N" : YNS(3)="X"
0260	162C	
0260	162C	REM *****
0260	162C	PNUM=VAL(PGMIDS)/10
027F	162E	REM LENGTH OF STRING RECEIVED FROM THE OMR....
027F	162E	HEADER = 21
0286	1630	RESPONSE= 450
0280	1632	RECORDLENGTH = HEADER + RESPONSE
0298	1634	
0298	1634	N.PROC = 71 ' NUMBER OF PROCEDURES FOR THIS FORM
029F	1636	N.DIAG.COL=4 ' NUMBER OF DX COLUMNS ON THIS FORM
02A6	1638	
02A6	1638	DATFILS = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
02AF	163C	BTIMES=TIMES 'SCAN START TIME
02B8	1640	
02B8	1640	REM *** ENCOUNTER FORM PROCEDURE TABLE ***
02B8	1640	F.NAMES="RACPROC." + PGMIDS
02C6	1644	
02C6	1644	OPEN F.NAMES FOR INPUT AS #3
02D7	1644	FOR I600=0 TO 125
02D0	1644	INPUT #3,PROCD\$(I600)
02F5	1646	IF PROCD\$(I600)="ZZZZZ" THEN GOTO 4
0311	1646	NEXT I600
0320	1646	4 CLOSE #3
0327	1646	
0327	1646	REM *** ENCOUNTER FORM DIAGNOSIS TABLE ***
0327	1646	F.NAMES="RACDIAG." + PGMIDS
0335	1646	
0335	1646	OPEN F.NAMES FOR INPUT AS #3
0346	1646	FOR I600 = 0 TO 225
034C	1646	INPUT #3,DIAGN.TAB\$(I600)
0364	1646	IF DIAGN.TAB\$(I600)="ZZZZZ" THEN GOTO 6
0380	1646	NEXT I600
0390	1646	6 CLOSE #3
0397	1646	
0397	1646	REM INCLUDE: 'UCACAMP.OPT' INCLUDE THE OUTP UCA VALIDATE TABLE
0397	1646	REM INCLUDE: 'UCACAMP.IPT' INCLUDE THE INP UCA VALIDATE TABLE
0397	1646	
0397	1646	REM \$PAGE

Offset	Data	Source Line
0397	1646	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
039C	1646	CLS
03A0	1646	GOSUB 7000 'PRINT SCREEN HEADING
03A5	1646	
03A5	1646	REM *****
03A5	1646	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
03A5	1646	REM *****
03A5	1646	REM
03A5	1646	OPEN DATFILS FOR APPEND AS #1
03B7	1646	
03B7	1646	REM *****
03B7	1646	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
03B7	1646	REM *****
03B7	1646	LPRINT CHR\$(15);
03C2	1646	WIDTH "LPT1:",160
03CC	1646	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
03D8	1648	COLOR 14
03DF	1648	LOCATE 11,26 : PRINT "EMERGENCY ROOM ENCOUNTER FORM "
03F4	1648	COLOR FORE,BACK,BORD
040A	1648	
040A	1648	REM *****
040A	1648	REM **** COMMUNICATIONS SETUP ****
040A	1648	REM *****
040A	1648	REM PROTOCOL
040A	1648	GOSUB 9001
040F	1648	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0429	164C	
0429	164C	REM START SCANNER (SI)
0429	164C	CNTRLOPT = 1 : GOSUB 9010
0435	164E	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
044F	164E	
044F	164E	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
0464	164E	READTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
0468	1650	
0468	1650	REM *****
0468	1650	REM *** SET SCAN SHEET CALL ***
0468	1650	REM *****
0468	1650	REM
0468	1650	
0468	1650	10 REM - RETURN POINT TO READ NEXT SHEET
046C	1650	
046C	1650	AS=INKEY\$
0475	1654	IF AS=CHR\$(27) THEN GOTO 25000
0488	1654	
0488	1654	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
0490	1654	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
04AC	1658	
04AC	1658	TEXT\$="" 'CLEAR THE INPUT AREA
04B5	1658	GOSUB 8000 'DECODE HEADER
04BA	1658	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
04BF	1658	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
04C4	1658	LITHO\$ = MID\$(TEXT\$,22,8)
04D6	165C	GOSUB 8070 'CHECK FOR SCANNER ERRORS
04D8	165C	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

04E0 165C
04E0 165C      REM $INCLUDE: 'RACM830.M01'  INCLUDE THE READ FORM REFORMAT/EDIT MOD
04E0 165C      REM *****
04E0 165C      REM      ****      AMBULATORY CARE INFORMATION SYSTEM      11 APR 87      ****
04E0 165C      REM      ****      D R BOLLING      ****
04E0 165C      REM      ****      MODULE NAME      :      RACREAD.MOD      ****
04E0 165C      REM      ****      SCANNER PROGRAM #      :      EMERGENCY RM FORM      ****
04E0 165C      REM      ****
04E0 165C      REM      ****      PURPOSE      :      REFORMAT/EDIT THE FORM      ****
04E0 165C      REM      ****
04E0 165C      REM      *****
04E0 165C      REM      **** RESERVED LINE NUMBERS 100-199      ****
04E0 165C      REM      *****
04E0 165C
04E0 165C      N.ERR  =0      'COUNTS THE NUMBER OF ERRORS
04E7 165E
04E7 165E      REM      *** LITHO CODE DONE IN BAS PROGRAM ***
04E7 165E      REM      *** CLINIC ID (PREFIX + CODE) ***
04E7 165E      100  CL1.CODS="BIYA"      'DEFAULT CLINIC CODE
04F0 1662
04F0 1662      REM VISIT DATE
04F0 1662      104  CMS=MID$(DATE$,1,2)      'CURRENT MONTH
0502 1666      YRS=MID$(DATE$,9,2)      'CURRENT YEAR
0514 166A      XS=MID$(TEXT$,35,4)      'MONTH AND DAY
0526 166E      IF LEFT$(XS,2)<=CMS THEN 105      'OK, USE THIS YEAR
0538 166E      YRS=RIGHT$(STR$(VAL(YRS)-1),2) 'USE LAST YEAR
0558 166E      105  VIDATES=YRS+XS
0569 1672      D1$=MID$(XS,3,1)
0578 1676      D2$=MID$(XS,4,1)
0580 167A      IF D1$=" " AND D2$<>" " THEN RT.5000=2 : GOTO 106
0588 167C      IF D1$<>" " AND D2$=" " THEN RT.5000=2 : GOTO 106
05E9 167C      'EDIT VISIT DATE
05E9 167C      CK.5000$=VIDATES$
05F2 1680      GOSUB 5000      'DATE CHECK
05F7 1680      IF RT.5000=0 THEN GOTO 107
0606 1680      106  N.ERR=N.ERR+1
060E 1680      ED.MSG$(N.ERR)="VISIT DATE" + DATEERR$(RT.5000)
0634 1680
0634 1680      REM      *** PRIMARY PROVIDER ***
0634 1680      107  PPROV.PFX$=MID$(TEXT$,39,1)
0646 1684      PPROV.NUM$=MID$(TEXT$,40,4)
0658 1688
0658 1688      REM      *** PATIENT SSN ***
0658 1688      108  SSN$ = MID$(TEXT$,44,9)
066A 168C
066A 168C      REM      *** FAMILY MEMBER PREF ***
066A 168C      110  FHEMP$=MID$(TEXT$,53,2)
067C 1690
067C 1690      REM      *** FORM NUMBER ***
067C 1690      FRMNS=LEFT$(PGMID$,2)
0688 1694
0688 1694      REM      *** VISIT COUNT ***
0688 1694      112  VCNT$="1"      'BY DEFAULT
0694 1698

```

Offset	Data	Source Line
0694	1698	REM *** PRIMARY PROVIDER TIME ***
0694	1698	113 X=VAL(MID\$(TEXT\$,56,2))
06AA	169A	PPROV.TIMS=PROVIDER.TIMES(X)
068C	169E	
068C	169E	REM *** SECONDARY PROVIDER ***
068C	169E	SPROV.PFX\$=MID\$(TEXT\$,58,1)
06CE	16A2	SPROV.NUMS=MID\$(TEXT\$,59,4)
06E0	16A6	IF SPROV.PFX\$+SPROV.NUMS = " " THEN SPBL=1 ELSE SPBL=0
0705	16A8	IF SPROV.PFX\$=" " AND SPROV.NUMS <> " " THEN 114
072B	16A8	IF SPROV.PFX\$<>" " AND SPROV.NUMS = " " THEN 114
0751	16A8	GOTO 115
0755	16A8	
0755	16A8	114 N.ERR = N.ERR + 1
0750	16A8	ED.MSG\$(N.ERR)="PROV 2 CODE MISSING PREFIX OR NUMBER"
0771	16A8	
0771	16A8	REM *** SECONDARY PROVIDER TIME ***
0771	16A8	115 X=VAL(MID\$(TEXT\$,63,2))
0787	16A8	SPROV.TIMS=PROVIDER.TIMES(X)
0799	16AC	REM IS THERE A TIME AND NO SEC PROV CODED?
0799	16AC	IF SPROV.TIMS <> "000" AND SPBL = 1 THEN 116 'GO IF YES
078C	16AC	REM IS THERE NO TIME AND A SEC PROV CODED?
078C	16AC	IF SPROV.TIMS = "000" AND SPBL = 0 THEN 117 'GO IF YES
07DF	16AC	GOTO 118
07E3	16AC	
07E3	16AC	116 N.ERR = N.ERR + 1
07EB	16AC	ED.MSG\$(N.ERR)="TIME CODED WITH NO SEC PROV CODED"
07FF	16AC	GOTO 118
0803	16AC	
0803	16AC	117 N.ERR = N.ERR + 1
080B	16AC	ED.MSG\$(N.ERR)="NO PROV 2 TIME"
081F	16AC	
081F	16AC	REM *** REASON FOR SECONDARY PROVIDER ***
081F	16AC	118 SPROV.REAS=MID\$(TEXT\$,65,1)
0831	1680	REM IS THERE A REASON AND NO SEC PROV CODED?
0831	1680	IF SPROV.REAS <> " " AND SPBL = 1 THEN 119 'GO IF YES
0854	1680	REM IS THERE NO REASON AND A SEC PROV CODED?
0854	1680	IF SPROV.REAS <> " " AND SPBL = 1 THEN 120 'GO IF YES
0877	1680	GOTO 122
087B	1680	
087B	1680	119 N.ERR = N.ERR + 1
0883	1680	ED.MSG\$(N.ERR)="REASON CODED WITH NO SEC PROV CODED"
0897	1680	GOTO 122
089B	1680	
089B	1680	120 N.ERR = N.ERR + 1
08A3	1680	ED.MSG\$(N.ERR)="NO PROV 2 REASON"
08B7	1680	
08B7	1680	REM *** IF NOT SCHEDULED ***
08B7	1680	122 NOTSC\$="3" 'DEFAULT IS 3 (EMERGENCY)
08C0	1684	X\$=MID\$(TEXT\$,66,1)
08D2	1684	IF X\$="1" THEN NOTSC\$="1" 'SCHEDULED (FOLLOW-UP)
08E9	1684	
08E9	1684	REM *** PLACE OF VISIT ***
08E9	1684	NOTCL\$="1" 'DEFAULT IS 1 (CLINIC)
08F2	1688	124 X\$=MID\$(TEXT\$,67,1)

Offset	Data	Source Line
0904	1688	IF XS="1" THEN NOTCLS="8" 'SEEN LAST 48 HOURS
091B	1688	
091B	1688	REM *** REFERRAL CODE ***
091B	1688	REF.CODS=" "
0924	168C	
0924	168C	REM *** JOB RELATED VISIT ***
0924	168C	130 RELAT.VISS="N"
092D	16C0	IF MIDS(TEXT\$,72,1)="1" THEN RELAT.VISS="Y"
094E	16C0	
094E	16C0	REM *** MIL ONLY DUTY ***
094E	16C0	132 MILDUTS=MIDS(TEXT\$,73,1)
0960	16C4	
0960	16C4	REM *** MIL ONLY QTRS ***
0960	16C4	134 MILQTRS=MIDS(TEXT\$,74,1)
0972	16C8	
0972	16C8	REM *** MIL ONLY PROF ***
0972	16C8	136 MILPROS=MIDS(TEXT\$,75,1)
0984	16CC	
0984	16CC	REM *** NOT AVAILABLE ***
0984	16CC	138 NAVAILS=MIDS(TEXT\$,76,1)
0996	16D0	
0996	16D0	REM *** SPEC PREASSIGNED CLINIC ***
0996	16D0	140 INP.STOS=MIDS(TEXT\$,77,9)
09A8	16D4	GOSUB 5700 'CONVERT ARRAY
09AD	16D4	SPE.BUFS=BUF.STOS 'UP TO 9 TWO DIGIT CODES
09B6	16D0	
09B6	16D0	REM *** SPEC PROGRAMS ***
09B6	16D0	REM *** READ AS ARRAY ***
09B6	16D0	142 INP.STOS=MIDS(TEXT\$,86,7)
09C8	16D0	GOSUB 5700 'CONVERT ARRAY
09CD	16D0	SPPROGS=BUF.STOS 'UP TO 7 TWO DIGIT CODES
09D6	16E0	
09D6	16E0	REM *** PROVIDER 2 ADDL PROC 1 ***
09D6	16E0	144 PR2PRC1S="N"
09DF	16E4	IF MIDS(TEXT\$,93,1)="1" THEN PR2PRC1S="Y"
0A00	16E4	
0A00	16E4	REM *** ADDITIONAL PROCEDURE 1 ***
0A00	16E4	146 ADDP1S=MIDS(TEXT\$,94,5)
0A12	16E8	
0A12	16E8	REM *** PROVIDER 2 ADDL PROC 2 ***
0A12	16E8	148 PR2PRC2S="N"
0A1B	16EC	IF MIDS(TEXT\$,99,1)="1" THEN PR2PRC2S="Y"
0A3C	16EC	
0A3C	16EC	REM *** ADDITIONAL PROCEDURE 2 ****
0A3C	16EC	150 ADDP2S=MIDS(TEXT\$,100,5)
0A4E	16F0	
0A4E	16F0	REM *** ADMITTED ***
0A4E	16F0	ADMIT\$="N"
0A57	16F4	IF MIDS(TEXT\$,105,1)="1" THEN ADMIT\$="Y"
0A78	16F4	
0A78	16F4	REM *** EDIT FOR MIL ONLY BOX ***
0A78	16F4	REM ... THE FOLLOWING FORMS TO NOT HAVE A MIL ONLY BOX
0A78	16F4	IF FRMNS="40" OR FRMNS="58" OR FRMNS="60" THEN 155
0AA8	16F4	IF FRMNS="63" OR FRMNS="67" OR FRMNS="76" THEN 155

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

0AD8 16F4      REM ... DO EDITS ON FMP=20 ONLY
0AD8 16F4      IF FMEMP$<>"20" THEN 155
0AE6 16F4      REM ... IS DUTY CODED
0AE6 16F4      IF MILDUT$<>" " THEN 152      'GO IF DUTY IS CODED
0AF4 16F4      REM ... IS QTRS CODED
0AF4 16F4      IF MILQTRS$<>" " THEN 152      'GO IF YES TO CHECK FOR ERROR
0B02 16F4      REM ... IS ADMITTED CODED
0B02 16F4      IF ADMITS$="Y" THEN 155      'IF YES THEN DO NOTHING
0B10 16F4      MILDUT$="1"                  'SET TO DUTY BY DEFAULT
0B19 16F4      GOTO 155
0B1D 16F4
0B1D 16F4      152 REM ... IS ADMITTED CODED
0B1E 16F4      IF ADMITS$="N" THEN 155      'OK
0B2C 16F4
0B2C 16F4      REM ... ERROR DUTY, QTRS AND ADMITTED CANNOT ALL BE CODED
0B2C 16F4      N.ERR = N.ERR + 1
0B34 16F4      ED.MSG$(N.ERR)="DUTY OR QTRS CANNOT BE CODED WITH ADMITTED"
0B48 16F4
0B48 16F4      REM *** UNLISTED PRIMARY DX ***
0B48 16F4      155 X=VAL(MID$(TEXT$,106,1))
0B5E 16F4      IF X=0 THEN PRIMDX$=""
0B72 16F8      IF X=1 THEN PRIMDX$="V"
0B86 16F8      IF X=2 THEN PRIMDX$="S"
0B9A 16F8      DXTMP$=MID$(TEXT$,107,5)
0BAC 16FC      IF DXTMP$=" " THEN GOTO 158
0B8E 16FC
0B8E 16FC      REM REMOVE LEADING BLANKS
0B8E 16FC      156 IF LEFT$(DXTMP$,1)=" " THEN DXTMP$=RIGHT$(DXTMP$,4)+" ":GOTO 156
0BED 16FC
0BED 16FC      158 LASTC$=RIGHT$(DXTMP$,1) 'GET LAST CHAR
0BFC 1700      IF LASTC$<>" " AND LASTC$<>"0" THEN PRIMDX$="" 'REMOVE V OR S
0C28 1700      PRIMDX$=LEFT$(PRIMDX$+DXTMP$,5) 'GET 5 CHAR ONLY
0C3A 1700
0C3A 1700      REM *** UNLISTED SECONDARY DX ***
0C3A 1700      160 X=VAL(MID$(TEXT$,112,1))
0C50 1700      IF X=0 THEN SECDX$=""
0C64 1704      IF X=1 THEN SECDX$="V"
0C78 1704      IF X=2 THEN SECDX$="S"
0C8C 1704      DXTMP$=MID$(TEXT$,113,5)
0C9E 1704      IF DXTMP$=" " THEN GOTO 164
0C80 1704
0C80 1704      REM REMOVE LEADING BLANKS
0C80 1704      162 IF LEFT$(DXTMP$,1)=" " THEN DXTMP$=RIGHT$(DXTMP$,4)+" ":GOTO 162
0CDF 1704
0CDF 1704      164 LASTC$=RIGHT$(DXTMP$,1) 'GET LAST CHAR
0CEE 1704      IF LASTC$<>" " AND LASTC$<>"0" THEN SECDX$="" 'REMOVE V OR S
0D1A 1704      SECDX$=LEFT$(SECDX$+DXTMP$,5) 'GET 5 CHAR ONLY
0D2C 1704
0D2C 1704      REM *** FOLLOW UP/RULE OUT ***
0D2C 1704      166 FU.ROS=MID$(TEXT$,118,1)
0D3E 1708
0D3E 1708      REM *** EVALUATION/SERV/PROC PROV 1 ***
0D3E 1708      168 INP.STOS=MID$(TEXT$,119,N.PROC)
0D51 1708      ESP.BUF1$=""

```

Offset	Data	Source Line
005A	170C	N.STO=1
0061	170E	
0061	170E	170 X.STO=INSTR(N.STO,INP.STO\$,"1")
0073	1710	IF X.STO=0 THEN GOTO 172 'THATS ALL
0082	1710	N.STO=X.STO + 1 'NEXT STARTING POINT
008A	1710	ESP.BUF1\$=ESP.BUF1\$ + PROCED\$(X.STO) 'ADD CODE TO BUFFER BY FIVES
009F	1710	IF N.STO <= LEN(INP.STO\$) THEN GOTO 170
0082	1710	
0082	1710	REM *** EVALUATION/SERV/PROC PROV 2 ***
0082	1710	172 INP.STO\$=MID\$(TEXT\$,119+N.PROC,N.PROC)
00C9	1710	ESP.BUF2\$=""
0002	1714	N.STO=1
0009	1714	
0009	1714	174 X.STO=INSTR(N.STO,INP.STO\$,"1")
00E8	1714	IF X.STO=0 THEN GOTO 176 'THATS ALL
00FA	1714	IF SPROV.NUM\$ = " " THEN 175 'NEED SEC PROV CODED
0E08	1714	N.STO=X.STO + 1 'NEXT STARTING POINT
0E10	1714	ESP.BUF2\$=ESP.BUF2\$ + PROCED\$(X.STO) 'ADD CODE TO BUFFER BY FIVES
0E25	1714	IF N.STO <= LEN(INP.STO\$) THEN GOTO 174
0E38	1714	GOTO 176
0E3C	1714	
0E3C	1714	175 N.ERR = N.ERR + 1
0E44	1714	ED.MSG\$(N.ERR)="PROC CODE FOR SEC PROV BUT NO SEC PROV CODED"
0E58	1714	
0E58	1714	176 T.POS = 119 + 2*N.PROC
0E66	1716	
0E66	1716	REM *** PRIMARY DX ***
0E66	1716	REM * IF OTHER PRIM DX IS CODED THEN SKIP THIS SECTION *
0E66	1716	186 IF PRIMDX\$<>" " THEN GOTO 192
0E7B	1716	X.POS=T.POS 'STARTING POSITION
0E7F	1718	C1\$IZ=55 : C3\$IZ=50 'NO. OF ITEMS IN EACH COL
0E8D	171C	C2\$IZ=63 : C4\$IZ=26
0E9B	1720	GOSUB 5800 'GET POSITION
0EA0	1720	IF X.FIN=0 THEN GOTO 188
0EAF	1722	IF RT.5800=0 THEN GOTO 190
0EBE	1724	N.ERR=N.ERR + 1
0EC6	1724	ED.MSG\$(N.ERR)="PRIMARY DX HAS MULTIPLE CODES"
0EDA	1724	GOTO 190
0EDE	1724	
0EDE	1724	188 N.ERR=N.ERR + 1
0EE6	1724	ED.MSG\$(N.ERR)="PRIMARY DX NOT CODED"
0EFA	1724	
0EFA	1724	190 PRIMDX\$=DIAGN.TAB\$(X.FIN)
0F0C	1724	
0F0C	1724	192 T.POS = T.POS + 2 * N.DIAG.COL
0F1B	1724	NDX = 55+63+50+26-1
0F22	1726	
0F22	1726	REM *** SECONDARY DX ***
0F22	1726	194 INP.STO\$=MID\$(TEXT\$,T.POS,NDX)
0F36	1726	DX2.BUF\$=""
0F4F	172A	N.STO=1
0F46	172A	
0F46	172A	196 X.STO=INSTR(N.STO,INP.STO\$,"1")
0F58	172A	IF X.STO=0 THEN GOTO 197 'THATS ALL

RACP830

EMERGENCY ROOM FORM DESTING/DECODE PROGRAM

PAGE 12

07-06-87

14:57:17

IBM Personal Computer BASIC Compiler V1.00

```

Offset Data Source Line
0F67 172A N.STO=X.STO + 1 'NEXT STARTING POINT
0F6F 172A DX2.BUF$=DX2.BUF$ + DIAGN.TAB$(X.STO) 'ADD CODE TO BUFFER BY FIVES
0F84 172A IF N.STO <= LEN(INP.STO$) THEN GOTO 196
0F97 172A 197 IF SECDX$="" " THEN GOTO 198
0FA9 172A DX2.BUF$=DX2.BUF$+SECDX$ 'ADD OTHER SEC DX IF THERE
0FB5 172A
0FB5 172A 198 T.POS = T.POS + NDX
0FC0 172A
0FC0 172A IF PGMID$ <> "740" THEN 199 'SKIP IF NOT PSYCHOLOGY
0FCE 172A REM *** PSYCHOMETRIC ASSESSMENTS ***
0FCE 172A INP.STO$=MID$(TEXT$,T.POS,4)
0FE1 172A GOSUB 5700 'CONVERT ARRAY
0FE6 172A PSYCHOS=BUF.STO$ 'UP TO 4 TWO DIGIT CODES
0FEF 172E
0FEF 172E 199 REM
0FF0 172E REM -----END OF MODULE RACM830.MO1-----
0FF0 172E
0FF0 172E IF N.ERR = 0 THEN GOTO 997
0FF0 172E LPRINT "LITHO # ";LITHO$;" ...ERRORS"
1011 172E FOR 1997 = 1 TO N.ERR
101E 1730 LPRINT USING "### ";1997;
102A 1732 LPRINT "=> ";ED.MSG$(1997)
1040 1732 NEXT 1997
1051 1732 LN.COUNT = LN.COUNT + N.ERR + 1
105D 1734 CNTRLOPT = 6
1064 1734 GOSUB 9010 'REJECT THE FORM
1069 1734 GOTO 998 'BYPASS THE DISK WRITER....
106D 1734
106D 1734 997 REM $INCLUDE: 'RACM830.MO2' REM INCLUDE THE BASE FORM DISK WRITER
106E 1734 REM *****
106E 1734 REM **** AMBULATORY CARE INFORMATION SYSTEM 11 APR 87 ****
106E 1734 REM **** D R BOLLING ****
106E 1734 REM **** MODULE NAME : RACM830.MO2 ****
106E 1734 REM **** SCANNER PROGRAM # : EMERGENCY ROOM FORM ****
106E 1734 REM ****
106E 1734 REM **** PURPOSE : CREATE AND WRITE THE DISK ****
106E 1734 REM **** RECORD FOR INPUT TO FOCUS ****
106E 1734 REM ****
106E 1734 REM *****
106E 1734 REM **** RESERVED LINE NUMBERS 200-299 ****
106E 1734 REM *****
106E 1734
106E 1734 REM BUILD THE OUTPUT RECORD
106E 1734
106E 1734 GOSUB 276 'BUILD THE RECORD KEY
1073 1734
1073 1734 REM *****
1073 1734 REM **** RECORD TYPE "1" - MAIN TRANSACTION ****
1073 1734 REM *****
1073 1734
1073 1734 GOSUB 278 'BUILD DATA FOR TYPE 1
1078 1734 RECOUNT$=PGMID$+"1"+RECKEY$+RECOD1$ 'TRANSACTION ID PLUS RECORD
1092 1740 GOSUB 280
1097 1740 PRINT #1,RECOUNT$

```

Offset	Data	Source Line
10A2	1740	
10A2	1740	REM *****
10A2	1740	REM **** RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE ****
10A2	1740	REM *****
10A2	1740	
10A2	1740	REM *** ADD ADDITIONAL PROCEDURES IF ANY ***
10A2	1740	IF ADDP1\$<>" " AND PR2PRC1\$="N" THEN ESP.BUF1\$=ESP.BUF1\$+ADDP1\$
1004	1740	IF ADDP2\$<>" " AND PR2PRC2\$="N" THEN ESP.BUF1\$=ESP.BUF1\$+ADDP2\$
1106	1740	
1106	1740	IF ADDP1\$<>" " AND PR2PRC1\$="Y" THEN ESP.BUF2\$=ESP.BUF2\$+ADDP1\$
1138	1740	IF ADDP2\$<>" " AND PR2PRC2\$="Y" THEN ESP.BUF2\$=ESP.BUF2\$+ADDP2\$
116A	1740	
116A	1740	REM *** PROCESS PROV 1 PROCEDURES ***
116A	1740	
116A	1740	IF LEN(ESP.BUF1\$)=0 THEN GOTO 206
117B	1740	
117B	1740	RPOINT=1
1182	1742	202 RECOD2\$=MID\$(ESP.BUF1\$,RPOINT,5)
1195	1746	IF RECOD2\$=" " THEN GOTO 204
11A7	1746	
11A7	1746	RECOUT\$=PGMID\$+"2"+RECKEY\$+"1"+RECOD2\$ 'TRANSACTION ID PLUS RECORD
11C7	1746	GOSUB 280
11CC	1746	PRINT #1,RECOUT\$
11D7	1746	
11D7	1746	204 RPOINT = RPOINT + 5
11E1	1746	IF RPOINT < LEN(ESP.BUF1\$) THEN GOTO 202
11F4	1746	
11F4	1746	REM *** PROCESS PROV 2 PROCEDURES ***
11F4	1746	
11F4	1746	206 IF LEN(ESP.BUF2\$)=0 THEN GOTO 212
1205	1746	
1205	1746	RPOINT=1
120C	1746	208 RECOD2\$=MID\$(ESP.BUF2\$,RPOINT,5)
121F	1746	IF RECOD2\$=" " THEN GOTO 210
1231	1746	
1231	1746	RECOUT\$=PGMID\$+"2"+RECKEY\$+"2"+RECOD2\$ 'TRANSACTION ID PLUS RECORD
1254	1746	GOSUB 280
1259	1746	PRINT #1,RECOUT\$
1264	1746	
1264	1746	210 RPOINT = RPOINT + 5
126E	1746	IF RPOINT < LEN(ESP.BUF2\$) THEN GOTO 208
1281	1746	
1281	1746	212 REM END OF TYPE 2 RECORDS
1282	1746	REM *****
1282	1746	REM **** RECORD TYPE "3" - RECKEY PLUS SPECIFIC PRE CLINIC CODES ****
1282	1746	REM *****
1262	1746	IF LEN(SPE.BUF\$)=0 THEN 218
128F	1746	
128F	1746	RPOINT=1
1296	1746	214 RECOD3\$=RIGHT\$(MID\$(SPE.BUF\$,RPOINT,2),1)
12AF	174A	IF RECOD3\$=" " THEN 216
12BD	174A	
12FD	174A	RECOUT\$=PGMID\$ + "3" + RECKEY\$ + RECOD3\$
12D7	174A	GOSUB 280

RACP830

EMERGENCY ROOM FORM DESTING/DECODE PROGRAM

PALE 14

07-06-87

14:57:17

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
12DC	174A	PRINT #1,RECOU\$
12E7	174A	
12E7	174A	216 RPOINT=RPOINT+2
12F0	174A	IF RPOINT < LEN(SPE.BUF\$) THEN 214
12FC	174A	
12FC	174A	218 REM END OF TYPE 3 RECORDS
12FD	174A	
12FD	174A	REM *****
12FD	174A	REM **** RECORD TYPE "4" - RECKEY PLUS OTHER CODES ****
12FD	174A	REM *****
12FD	174A	
12FD	174A	IF LEN(OTH.BUF\$)=0 THEN GOTO 224
130E	174E	
130E	174E	RPOINT=1
1315	174E	220 RECOD4\$=MID\$(OTH.BUF\$,RPOINT,5)
1328	1752	IF RECOD4\$="" THEN GOTO 222
133A	1752	
133A	1752	RECOU\$=PGMID\$+"4"+RECKEY\$+RECOD4\$ 'TRANSACTION ID PLUS RECORD
1354	1752	GOSUB 280
1359	1752	PRINT #1,RECOU\$
1364	1752	
1364	1752	222 RPOINT = RPOINT + 5
136E	1752	IF RPOINT < LEN(OTH.BUF\$) THEN GOTO 220
1381	1752	
1381	1752	
1381	1752	224 REM END OF TYPE 4 RECORDS
1382	1752	
1382	1752	REM *****
1382	1752	REM **** RECORD TYPE "5" - RECKEY PLUS GROUP I DATA ****
1382	1752	REM *****
1382	1752	
1382	1752	REM *****
1382	1752	REM **** RECORD TYPE "6" - RECKEY PLUS SPECIAL PROGRAMS ****
1382	1752	REM *****
1382	1752	IF LEN(SPPROG\$)=0 THEN 262
138F	1752	
138F	1752	RPOINT=1
1396	1752	258 RECOD6\$=RIGHT\$(MID\$(SPPROG\$,RPOINT,2),1)
13AF	1756	IF RECOD6\$="" THEN 260
13BD	1756	
13BD	1756	RECOU\$=PGMID\$ + "6" + RECKEY\$ + RECOD6\$
13D7	1756	GOSUB 280
13DC	1756	PRINT #1,RECOU\$
13E7	1756	
13E7	1756	260 RPOINT=RPOINT+2
13F0	1756	IF RPOINT < LEN(SPPROG\$) THEN 258
13FC	1756	
13FC	1756	262 REM END OF TYPE 6 RECORDS
13FD	1756	
13FD	1756	REM *****
13FD	1756	REM **** RECORD TYPE "7" - RECKEY PLUS SECOND DX CODE ****
13FD	1756	REM *****
13FD	1756	
13FD	1756	IF LEN(DX2.BUF\$)=0 THEN GOTO 268

Offset Date Source Line

```

140E 1756
140E 1756      RPOINT=1
1415 1756      264 RECOD7$=MID$(DX2.BUF$,RPOINT,5)
1428 175A      IF RECOD7$="" THEN GOTO 266
143A 175A
143A 175A      RECOD7$=PGMID$+"7"+RECKEY$+RECOD7$ 'TRANSACTION ID 7
1454 175A      GOSUB 280
1459 175A      PRINT #1,RECOD7$
1464 175A
1464 175A      266 RPOINT = RPOINT + 5
146E 175A      IF RPOINT < LEN(DX2.BUF$) THEN GOTO 264
1481 175A
1481 175A      268 REM END OF TYPE 7 RECORDS
1482 175A
1482 175A
1482 175A      GOTO 299
1486 175A
1486 175A      REM *****
1486 175A      REM **** SUBROUTINE 270 - BUILD THE RECORD KEY ****
1486 175A      REM *****
1486 175A      276 RECKEY$=""
148F 175A
148F 175A      REM *** CLINIC ID (PREFIX + COD) ***
148F 175A      RECKEY$= CL1.COD$
1498 175A
1498 175A      REM *** VISIT DATE ***
1498 175A      RECKEY$=RECKEY$+ VDATE$
14A4 175A
14A4 175A      REM *** PRIMARY PROVIDER ***
14A4 175A      RECKEY$ = RECKEY$ + PPROV.PFX$ + PPROV.NUM$
14B7 175A
14B7 175A      REM *** PATIENT SSN ***
14B7 175A      RECKEY$ = RECKEY$ + SSN$
14C3 175A
14C3 175A      REM *** FAMILY MEMBER PREF ***
14C3 175A      RECKEY$ = RECKEY$ + FMEMP$
14CF 175A
14CF 175A      REM *** LITHO CODE ***
14CF 175A      RECKEY$ = RECKEY$ + LITHO$
14DB 175A
14DB 175A      REM *** FORM NUMBER ***
14DB 175A      RECKEY$ = RECKEY$ + FRMN$
14E7 175A
14E7 175A      RETURN
14EA 175A
14EA 175A      REM *****
14EA 175A      REM **** SUBROUTINE 272 - BUILD THE DATA FOR TYPE 1 ****
14EA 175A      REM *****
14EA 175A      278 RECOD1$=""
14F3 175A
14F3 175A      REM *** VISIT COUNT ***
14F3 175A      RECOD1$ = RECOD1$ + VCNT$
14FF 175A
14FF 175A      REM *** PRIMARY PROV TIME ***

```

RACP830

EMERGENCY ROOM FORM DESTURING/DECODE PROGRAM

PACF 16

07-06-87

14:57:17

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
14FF	175A	RECOD1\$ = RECOD1\$ + PPROV.TIMS
150B	175A	
150B	175A	REM *** SECONDARY PROVIDER ***
150B	175A	RECOD1\$ = RECOD1\$ + SPROV.PFX\$ + SPROV.NUM\$
151E	175A	
151E	175A	REM *** SECONDARY PROVIDER TIME ***
151E	175A	RECOD1\$ = RECOD1\$ + SPROV.TIMS
152A	175A	
152A	175A	REM *** REASON FOR SECONDARY PROVIDER ***
152A	175A	RECOD1\$ = RECOD1\$ + SPROV.REAS
1536	175A	
1536	175A	REM *** APPORINTMENT STATUS ***
1536	175A	RECOD1\$ = RECOD1\$ + NOTSC\$
1542	175A	
1542	175A	REM *** REFERRAL CODE ***
1542	175A	RECOD1\$ = RECOD1\$ + REF.PFX\$ + REF.COD\$
1555	175E	
1555	175E	REM *** PLACE OF VISIT ***
1555	175E	RECOD1\$ = RECOD1\$ + NOTCL\$
1561	175E	
1561	175E	REM *** JOB RELATED VISIT ***
1561	175E	RECOD1\$ = RECOD1\$ + RELAT.VISS
1560	175E	
1560	175E	REM *** MIL ONLY DUTY ***
1560	175E	RECOD1\$ = RECOD1\$ + MILDUT\$
1579	175E	
1579	175E	REM *** MIL ONLY QTRS ***
1579	175E	RECOD1\$ = RECOD1\$ + MILQTR\$
1585	175E	
1585	175E	REM *** MIL ONLY PROFILE ***
1585	175E	RECOD1\$ = RECOD1\$ + MILPRO\$
1591	175E	
1591	175E	REM *** NOT AVAILABLE ***
1591	175E	RECOD1\$ = RECOD1\$ + NAVAIL\$
1590	175E	
1590	175E	REM *** ADMITTED ***
1590	175E	RECOD1\$ = RECOD1\$ + ADMIT\$
15A9	175E	
15A9	175E	REM *** INFIELD (NOT ON GENERAL FORM) ***
15A9	175E	RECOD1\$ = RECOD1\$ + " "
15B5	175E	
15B5	175E	REM *** INJURY (NOT ON GENERAL FORM) ***
15B5	175E	RECOD1\$ = RECOD1\$ + " "
15C1	175E	
15C1	175E	REM *** PURPOSE OF VISIT (NOT ON GENERAL FORM) ***
15C1	175E	RECOD1\$ = RECOD1\$ + " "
15CD	175E	
15CD	175E	REM *** PRIM FOLLOW-UP/RULE OUT ***
15CD	175E	RECOD1\$ = RECOD1\$ + FU.RO\$
15D9	175E	
15D9	175E	REM *** PRIMARY DX CODE ***
15D9	175E	RECOD1\$ = RECOD1\$ + PRIMDX\$
15E5	175E	
15E5	175E	RETURN

Offset Data Source Line

```

15E8 175E
15E8 175E REM *****
15E8 175E REM **** SUBROUTINE 230 - PAD THE RECORD TO MAXLENGTH ****
15E8 175E REM *****
15E8 175E 280 PAD=MAXLENGTH - LEN(RECOUT$) 'FIND OUT HOW SHORT THE RECORD IS
15F6 1760 RECOUT$ = RECOUT$ + STRING$(PAD,PAD$) 'PAD THE RECORD WITH FILL CHAR
1609 1760 RETURN
160C 1760
160C 1760 299 REM
160D 1760
160D 1760 REM -----END OF MODULE RACM830.M02-----
160D 1760
160D 1760 998 REM CONTINUE
160E 1760
160E 1760 999 READTYPE = 2
1615 1760 IF LNLCOUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
1625 1762 GOTO 10
1629 1762
1629 1762 REM END OF SCAN/DECODE/WRITE LOOP =====
1629 1762
1629 1762 1000 REM $INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
162A 1762 REM *****
162A 1762 REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
162A 1762 REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
162A 1762 REM *****
162A 1762 REM PATIENT OMR INPUT PROGRAM *
162A 1762 REM *
162A 1762 REM This program verifies user is logged on properly. If there is no *
162A 1762 REM valid user logged on at the time of execution, this subroutine will*
162A 1762 REM chain to the logon program RACP05, otherwise a return is issued. *
162A 1762 REM *****
162A 1762 REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
162A 1762 REM *****
162A 1762 1001 OPEN "I",1,"RACLOG.DAT"
163C 1762 IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
164A 1762 INPUT #1,USER$(1),DT$,TM$,PID$
166B 176E IF USER$(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
1579 176E IF USER$(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
1687 176E CLOSE 1
168E 176E SCREEN 0,1,0,0
16A4 176E COLOR FORE,BACK,BORD
16BA 176E CLS
16BE 176E RETURN
16C1 176E
16C1 176E 1002 CLOSE
16C5 176E CHAIN "RACP05"
16CC 176E '=====END OF LOGON VERIFY SUBROUTINE 1000=====
16CC 176E
16CC 176E 2000 REM $INCLUDE: 'RACS2000.SUB' INCLUDE THE RIPLY/DELAY SUB
16CD 176E REM *****
16CD 176E REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
16CD 176E REM **** SKIP COLE ****
16CD 176E REM **** SUBROUTINE NAME : RACS2000.SUB ****
16CD 176E REM **** SCANNER PROGRAM # : ALL ****

```

RACPB30

EMERGENCY ROOM FORM DESTING/DECODE PROGRAM

PAGE 18

07-06-87

14:57:17

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
16CD	176E	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
16CD	176E	REM **** SERVERS AS A WAIT AND REPLY ****
16CD	176E	REM **** ENTRY MODULE ****
16CD	176E	REM **** INPUT : SINGLE KEYBOARD ENTRY ****
16CD	176E	REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
16CD	176E	REM **** RESERVED LINE ****
16CD	176E	REM **** NUMBERS : 2001-2010 ****
16CD	176E	REM *****
16CD	176E	2001 REM REPLY FUNCTION
16CE	176E	2002 REPLY\$=INKEY\$: IF REPLY\$="" THEN 2002
16E2	1772	REPLY=ASC(REPLY\$)
16EC	1774	IF REPLY > 90 THEN REPLY\$=CHR\$(REPLY XOR 32) 'CONVERT TO CAPS
1707	1774	IF REPLY\$ < "A" OR REPLY\$ > "Z" THEN REPLY\$="?"
1733	1774	RETURN
1736	1774	5000 REM \$INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB
1737	1774	REM *****
1737	1774	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1737	1774	REM **** SKIP COLE ****
1737	1774	REM **** SUBROUTINE NAME : RXXS5000.SUB ****
1737	1774	REM **** SCANNER PROGRAM # : ALL ****
1737	1774	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1737	1774	REM **** PERFORMS A DATE EDIT ****
1737	1774	REM **** INPUT : DATE TO BE CHECKED MUST BE ****
1737	1774	REM **** IN THE VARIABLE NAMED ****
1737	1774	REM **** 'CK.5000\$' ****
1737	1774	REM **** IN THE FORMAT "YYMMDD" ****
1737	1774	REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
1737	1774	REM **** VARIABLE. IF THIS VARIABLE ****
1737	1774	REM **** CONTAINS ANY NUMBER OTHER ****
1737	1774	REM **** THAN 0, AN ERROR WAS FOUND ****
1737	1774	REM **** IN THE DATE. ****
1737	1774	REM **** RESERVED LINE ****
1737	1774	REM **** NUMBERS : 5001-5009 ****
1737	1774	REM *****
1737	1774	RT.5000 = 0
173E	1774	CKYEAR = VAL(LEFT\$(CK.5000\$,2)) 'YEAR NUMERIC VALUE
1751	1776	CKMONTH = VAL(MID\$(CK.5000\$,3,2)) 'MONTH NUMERIC VALUE
1767	1778	CKDAY = VAL(RIGHT\$(CK.5000\$,2)) 'DAY NUMERIC VALUE
177A	177A	
177A	177A	IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
1790	177A	IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
17A6	177A	IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
17BC	177A	IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
17D2	177A	
17D2	177A	REM LEAP YEAR CHECK
17D2	177A	MOLENGTH(2) = 28
17D9	177A	IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY
17E8	177A	IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR

```

Offset  Data    Source Line
17FD 177A      MOLENGTH(2) = 29
1804 177A
1804 177A      5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
1823 177A
1823 177A      5009 RETURN
1826 177A
1826 177A      REM -----END OF SUBROUTINE 5000 -----
1826 177A
1826 177A      5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTP UCA VALIDATE SUB
1827 177A      5600 REM INCLUDE: 'RACS5600.SUB' INCLUDE THE INP UCA VALIDATE SUB
1828 177A      5700 REM $INCLUDE: 'RACS5700.SUB' INCLUDE MAP ONES TO POSITION NO.
1829 177A      REM *****
1829 177A      REM ****      AMBULATORY CARE DATA BASE          29 JUL 85      ****
1829 177A      REM ****                                  D R BOLLING      ****
1829 177A      REM ****      SUBROUTINE NAME      :      RXXS5700.SUB      ****
1829 177A      REM ****      SCANNER PROGRAM #      :      ALL      ****
1829 177A      REM ****      FUNCTION      :      THIS SUBROUTINE MODULE      ****
1829 177A      REM ****                                  CONVERTS A BINARY ARRAY INTO      ****
1829 177A      REM ****                                  TWO CHAR CODES.      ****
1829 177A      REM ****
1829 177A      REM ****      INPUT      :      INP.STO$ AS STRING      ****
1829 177A      REM ****
1829 177A      REM ****
1829 177A      REM ****      OUTPUT      :      BUF.STO$ AS STRING.      ****
1829 177A      REM ****
1829 177A      REM ****
1829 177A      REM ****      RESERVED LINE      ****
1829 177A      REM ****      NUMBERS      :      5710-5730      ****
1829 177A      REM ****
1829 177A      REM *****
1829 177A
1829 177A      BUF.STO$=""
1832 177A      N.STO=1
1839 177A      5710 X.STO=INSTR(N.STO,INP.STO$,"1")
1848 177A      IF X.STO=0 THEN GOTO 5720 'THATS ALL
185A 177A      N.STO = X.STO + 1      'NEXT STARTING POINT
1862 177A      X.STO = X.STO + 100      'PAD WITH LEADING ZERO
186C 177A      BUF.STO$ = BUF.STO$ + RIGHT$(STR$(X.STO),2)
1882 177A      IF N.STO <= LEN(INP.STO$) THEN GOTO 5710
1895 177A
1895 177A      5720 RETURN
1898 177A
1898 177A      REM ----- END OF SUBROUTINE 5700 -----
1898 177A      5800 REM $INCLUDE: 'RACS5800.SUB' INCLUDE FOUR COL DX CONVERTER
1899 177A      REM *****
1899 177A      REM ***      AMBULATORY CARE DATA BASE          30 JUL 85      ***
1899 177A      REM ***                                  D R BOLLING      ***
1899 177A      REM ***      SUBROUTINE NAME      :      RXXS5800.SUB      ***
1899 177A      REM ***      SCANNER PROGRAM      :      AS APPROPRIATE      ***
1899 177A      REM ***      FUNCTION      :      THIS SUBROUTINE MODULE READS      ***
1899 177A      REM ***                                  FOUR COLUMNS IN DX AREA AND      ***
1899 177A      REM ***                                  CONVERTS TO A POSITION IN A      ***
1899 177A      REM ***                                  TABLE. AN ERROR CODE IS      ***
1899 177A      REM ***
1899 177A      REM ***
1899 177A      REM ***      RETURNED IF MULTIPLE CODES ARE      ***

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

1899 177A REM *** PRESENT. ***
1899 177A REM *** ***
1899 177A REM *** INPUT : X.POS - STARTING POSITION IN ***
1899 177A REM *** STRING ***
1899 177A REM *** ***
1899 177A REM *** OUTPUT : X.FIN - TABLE POSITION OF DX ***
1899 177A REM *** RT.5800 = 1 ON ERROR ***
1899 177A REM *** ***
1899 177A REM *** RESERVED LINE NUMBERS : 5801 - 5899 ***
1899 177A REM *****
1899 177A RT.5800=0 'INITIALIZE ERROR INDICATOR
18A0 177A X.FIN=0 'INITIALIZE TABLE OFFSET RESULT
18A7 177A
18A7 177A REM ** COLUMN 1 **
18A7 177A X=0 'STARTING POINTER
18AE 177A X.SIZ=C1SIZ
18B5 177C X.STO=VAL(MID$(TEXT$,X.POS,2))
18CC 177C IF X.STO=0 THEN GOTO 5802
18DB 177C X.FIN=X.STO
18E2 177C
18E2 177C 5802 REM ** COLUMN 2 **
18E3 177C IF C2SIZ=0 THEN GOTO 5804
18F2 177C X=X+X.SIZ
18FD 177C X.POS=X.POS+2
1906 177C X.SIZ=C2SIZ
190D 177C X.STO=VAL(MID$(TEXT$,X.POS,2))
1924 177C IF X.STO=0 THEN GOTO 5804
1933 177C IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
1942 177C X.FIN=X.STO + X
194D 177C
194D 177C 5804 REM ** COLUMN 3 **
194E 177C IF C3SIZ=0 THEN GOTO 5806
195D 177C X=X+X.SIZ
1968 177C X.POS=X.POS+2
1971 177C X.SIZ=C3SIZ
1978 177C X.STO=VAL(MID$(TEXT$,X.POS,2))
198F 177C IF X.STO=0 THEN GOTO 5806
199E 177C IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
19AD 177C X.FIN=X.STO + X
19B8 177C
19B8 177C 5806 REM ** COLUMN 4 **
19B9 177C IF C4SIZ=0 THEN GOTO 5808
19C8 177C X=X+X.SIZ
19D3 177C X.POS=X.POS+2
19DC 177C X.SIZ=C4SIZ
19E3 177C X.STO=VAL(MID$(TEXT$,X.POS,2))
19FA 177C IF X.STO=0 THEN GOTO 5808
1A09 177C IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
1A18 177C X.FIN=X.STO + X
1A23 177C
1A23 177C 5808 REM
1A24 177C GOTO 5899
1A28 177C
1A28 177C 5890 RT.5800=1 'ERROR - MULTIPLE CODES

```

```

Offset  Data   Source Line
-----
1A2F  177C
1A2F  177C  5899 RETURN
1A32  177C
1A32  177C  REM -----END OF SUBROUTINE RXXS5800.SUB-----
1A32  177C
1A32  177C
1A32  177C  6000 REM $INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE SUB
1A33  177C  REM *****
1A33  177C  REM ****    AMBULATORY CARE DATA BASE          13 APR 85    ****
1A33  177C  REM ****                                SKIP COLE          ****
1A33  177C  REM ****    SUBROUTINE NAME      :    RXXS6000.SUB          ****
1A33  177C  REM ****    SCANNER PROGRAM #    :    ALL                    ****
1A33  177C  REM ****    FUNCTION              :    THIS SUBROUTINE MODULE ****
1A33  177C  REM ****                                PERFORMS INSTRING SEARCH ****
1A33  177C  REM ****
1A33  177C  REM ****    INPUT                  :    STRING TO BE SEARCHED MUST ****
1A33  177C  REM ****                                BE IN THE VARIABLE NAMED : ****
1A33  177C  REM ****                                'X$'                  ****
1A33  177C  REM ****
1A33  177C  REM ****    OUTPUT                  :    'TOT' = TOTAL NUMBER OF ****
1A33  177C  REM ****                                HITS IN THE DESTING ****
1A33  177C  REM ****                                'HOLD$( )' IS THE ARRAY ****
1A33  177C  REM ****                                CONTAINING THE NUMERIC ****
1A33  177C  REM ****                                VALUE OF THE HIT POSITIONS ****
1A33  177C  REM ****
1A33  177C  REM ****
1A33  177C  REM ****    RESERVED LINE ****
1A33  177C  REM ****            NUMBERS      :    6001-6009          ****
1A33  177C  REM ****
1A33  177C  6001      PTR = INSTR(X$, "1")
1A41  177E      TOT = 0
1A48  1780      WHILE PTR > 0
1A53  1780          TOT=TOT+1
1A58  1780          HOLD$(TOT) = RIGHT$(STR$(PTR),2)
1A7C  1780          PTR=PTR+1
1A85  1780          PTR = INSTR(PTR,X$, "1")
1A97  1780      WEND
1A98  1780      RETURN
1A9E  1780  REM -----END OF SUBROUTINE RXXS6000.SUB-----
1A9E  1780
1A9E  1780
1A9E  1780  7000 REM $INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
1A9F  1780  REM *****
1A9F  1780  REM ****    AMBULATORY CARE DATA BASE          13 APR 85    ****
1A9F  1780  REM ****                                SKIP COLE          ****
1A9F  1780  REM ****    SUBROUTINE NAME      :    RACS7000.SUB          ****
1A9F  1780  REM ****    SCANNER PROGRAM #    :    ALL                    ****
1A9F  1780  REM ****    FUNCTION              :    THIS SUBROUTINE MODULE ****
1A9F  1780  REM ****                                PRINTS THE STANDARD SCREEN ****
1A9F  1780  REM ****                                HEADING.          ****
1A9F  1780  REM ****    INPUT                  :    COMMON VARIABLE USERS(2) ****
1A9F  1780  REM ****                                SYSTEM DATE          ****
1A9F  1780  REM ****

```

Offset Data Source Line IBM Personal Computer BASIC Compiler V1.50

```

1A9F 1780 REM **** OUTPUT : SCREEN HEADING ****
1A9F 1780 REM **** ****
1A9F 1780 REM **** RESERVED LINE ****
1A9F 1780 REM **** NUMBERS : 7001-7010 ****
1A9F 1780 REM ****
1A9F 1780
1A9F 1780 7001 LOCATE 1,1
1AA9 1780 PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
1AB1 1780 LOCATE 1,65
1ABE 1780 PRINT DATES;
1AC6 1780 LOCATE 2,1
1AD3 1780 PRINT "USER : ";USER$(1)
1AE0 1780 RETURN
1AE3 1780
1AE3 1780 7100 REM $INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
1AE4 1780 REM *****
1AE4 1780 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1AE4 1780 REM **** SKIP COLE ****
1AE4 1780 REM **** SUBROUTINE NAME : RXXS7100.SUB ****
1AE4 1780 REM **** SCANNER PROGRAM # : ALL ****
1AE4 1780 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1AE4 1780 REM **** PRINTS THE STANDARD HEADING ****
1AE4 1780 REM **** ON THE PRINTER. ****
1AE4 1780 REM **** INPUT : DATE,PAGE,PGMIDS,PGMTITLS ****
1AE4 1780 REM **** ****
1AE4 1780 REM **** OUTPUT : PRINTER HEADING, LN.COUNT ****
1AE4 1780 REM **** ****
1AE4 1780 REM **** RESERVED LINE ****
1AE4 1780 REM **** NUMBERS : 7101-7110 ****
1AE4 1780 REM ****
1AE4 1780
1AE4 1780 7101 IF PAGE > 0 THEN LPRINT CHR$(12);
1AFA 1780 LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITLS;
1B07 1784 LPRINT TAB(70);DATES
1B1A 1784 PAGE=PAGE+1
1B22 1784 LPRINT "PROGRAM ";PGMIDS;TAB(70);"PAGE";
1B3F 1784 LPRINT USING "####";PAGE
1B4B 1784 LPRINT
1B53 1784 LN.COUNT=3
1B5A 1784 RETURN
1B5D 1784
1B5D 1784 8000 REM $INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
1B5E 1784 REM *****
1B5E 1784 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ***
1B5E 1784 REM **** SKIP COLE ***
1B5E 1784 REM **** SUBROUTINE NAME : RXXS8000.SUB ***
1B5E 1784 REM **** SCANNER PROGRAM # : ALL ***
1B5E 1784 REM **** FUNCTION : THIS SUBROUTINE MODULE ***
1B5E 1784 REM **** IS A GROUPING THAT PERFORMS ***
1B5E 1784 REM **** VARIOUS DECODING FUNCTIONS ***
1B5E 1784 REM **** ON THE SCANNER DATA ***
1B5E 1784 REM ****
1B5E 1784 REM **** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
1B5E 1784 REM **** 8050 - CHECK FOR END OF JOB ***

```

```

Offset  Data  Source Line
185E 1784 REM **** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
185E 1784 REM **** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
185E 1784 REM **** (RETURNED IN TEXT$ STRING VARIABLE) ***
185E 1784 REM **** ***
185E 1784 REM **** INPUT : SHEET RECORD, RECORD LENGTH ***
185E 1784 REM **** ***
185E 1784 REM **** OUTPUT : 'TEXT$' TRING VARIABLE ***
185E 1784 REM **** ***
185E 1784 REM **** RESERVED LINE ***
185E 1784 REM **** NUMBERS : 8001-8500 ***
185E 1784 REM *****
185E 1784
185E 1784 'DECODE THE HEADER ONLY
185E 1784 8001 POINTER = 0
1865 1786 RECORDPTR = VARPTR(SHEETREC(0))
186C 1788 FOR J8000 = 1 TO 21
1873 1788 8002 TEXT$= TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
1891 1788 POINTER=POINTER+1
1899 1788 NEXT J8000
18A8 178A PROGRAM$= LEFT$(TEXT$,3)
18B7 178E BATCH$= MID$(TEXT$,4,3)
18C9 1792 SERIAL$= MID$(TEXT$,7,4)
18D8 1796 RUNID$= MID$(TEXT$,11,1)
18ED 179A FORM$= MID$(TEXT$,12,2)
18FF 179E POCKET$= MID$(TEXT$,14,1)
1C11 17A2 SCANERR1$=MID$(TEXT$,16,2)
1C23 17A6 SCANERR2$=MID$(TEXT$,18,2)
1C35 17AA SCANERR3$=MID$(TEXT$,20,2)
1C47 17AE GOTO 8500
1C4B 17AE
1C4B 17AE 8050 REM CHECK FOR END OF JOB/END OF BATCH
1C4C 17AE IF PROGRAM$ = PGMID$ THEN GOTO 8500
1C5E 17AE LPRINT STRING$(80,"")
1C6C 17AE LPRINT
1C74 17AE LPRINT "RECORDS PROCESSED ... ";SERIAL$
1C81 17AE LPRINT "STARTED AT ..... ";BTIM$
1C8E 17AE LPRINT "ENDED AT ..... ";TIM$
1C9B 17AE LPRINT CHR$(12)
1CA6 17AE GOTO 30000
1CAA 17AE
1CAA 17AE 8070 REM CHECK FOR SCANNER ERRORS
1CAB 17AE IF POCKET$ = " " GOTO 8500
1CBO 17AE LPRINT LITHOS;
1CC5 17AE LPRINT " ... SCANNER ERRORS : ";
1CCD 17AE LPRINT SCANERR1$;" / ";
1CDA 17AE LPRINT SCANERR2$;" / ";
1CE7 17AE LPRINT SCANERR3$
1CEF 17AE LN=LN+1
1CF7 17B0 GOTO 999
1CFB 17B0
1CFB 17B0 8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
1CFC 17B0 LOCATE 5,1:PRINT "PROGRAM ";PROGRAM$;
1D16 17B0
1D16 17B0 PRINT " BATCH ";BATCH$;

```

PAGE 24
07-06-87
14:57:17

Offset	Data	Source Line
--------	------	-------------

```

1023 1780          PRINT " RUN ";RUNIDS;
1030 1780          PRINT " FORM ";FORMS;
1030 1780          PRINT " POCKET ";POCKETS
104A 1780          GOTO 8500
104E 1780
104E 1780 8200 REM DECODE THE RESPONSE POSITIONS
104F 1780          POINTER = 21
1056 1780          RECORDPTR = VARPTR(SHEETREC(0))
1050 1780          FOR J8000 = 22 TO RECORDLENGTH
106A 1782 8202          TEXT$ = TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
1088 1782          POINTER=POINTER+1
1090 1782          NEXT J8000
10A1 1782
10A1 1782 8500 RETURN
10A4 1782
10A4 1782 REM ----- END OF RXXS8000.SUB -----
10A4 1782
10A4 1782 9000 REM $INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
10A5 1782 REM *****
10A5 1782 REM ***** AMBULATORY CARE DATA BASE 13 APR 85 *****
10A5 1782 REM ***** SKIP COLE *****
10A5 1782 REM ***** PROGRAM NAME : RACS9000.SUB *****
10A5 1782 REM ***** SCANNER PROGRAM # : ALL *****
10A5 1782 REM ***** FUNCTION : THIS SUBROUTINE MODULE *****
10A5 1782 REM ***** CONTROLS THE SCANNER I/O *****
10A5 1782 REM *****
10A5 1782 REM ***** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS *****
10A5 1782 REM ***** COMMUNICATIONS MANUAL AND THE *****
10A5 1782 REM ***** PRE-RELEASED SOFTWARE GUIDE *****
10A5 1782 REM *****
10A5 1782 REM *****
10A5 1782 REM ***** RESERVED LINE *****
10A5 1782 REM ***** NUMBERS : 9001-9100 *****
10A5 1782 REM *****
10A5 1782 REM *****
10A5 1782 REM *****
10A5 1782 REM ***** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER *****
10A5 1782 REM ***** ARGUMENTS: PRESET ... SEE BELOW *****
10A5 1782 REM *****
10A5 1782 9001 REM
10A6 1782          PROTOCOL(0) = 9600          'BAUD RATE
10A0 1782          PROTOCOL(1) = 78          'PARITY (SEE PAGE 4-8 OF MANUAL)
10B4 1782          PROTOCOL(2) = 8          'DATA BITS
10B8 1782          PROTOCOL(3) = 1          'STOP BITS
10C2 1782          PROTOCOL(4) = 2          'RS-232 PORT
10C9 1782          PROTOCOL(5) = 0          'WRITE TIME-OUT
10D0 1782          PROTOCOL(6) = 0          'READ TIME-OUT
10D7 1782
10D7 1782          ERRSTAT$ = SPACE$(60)
10E3 1782          ARGPTR = VARPTR(PROTOCOL(0))
10EA 1782          CALL SETUP (ARGPTR,ERRSTAT$)
10FB 1782          ERRMSG$=""
1F04 1782          IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="SETUP ERROR "+ERRSTAT$

```

Offset	Data	Source Line
1E20	17B4	GOTO 9100
1E24	17B4	
1E24	17B4	REM *****
1E24	17B4	REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
1E24	17B4	REM **** ARGUMENTS: CNTRLOPT ****
1E24	17B4	REM **** CNTRLOPT = 1 = START SCANNER (SI) ****
1E24	17B4	REM **** CNTRLOPT = 2 = STOP SCANNER (SO) ****
1E24	17B4	REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
1E24	17B4	REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
1E24	17B4	REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
1E24	17B4	REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
1E24	17B4	REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
1E24	17B4	REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
1E24	17B4	REM *****
1E24	17B4	9010 REM
1E25	17B4	ERRSTAT\$ = SPACE\$(60)
1E31	17B4	CALL CNTRLOPT (CNTRLOPT,ERRSTAT\$)
1E42	17B4	ERRMSG\$=""
1E48	17B4	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
1E67	17B4	GOTO 9100
1E6B	17B4	
1E6B	17B4	REM *****
1E6B	17B4	REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
1E6B	17B4	REM ****
1E6B	17B4	REM **** ARGUMENTS: READTYPE ****
1E6B	17B4	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
1E6B	17B4	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
1E6B	17B4	REM ****
1E6B	17B4	REM **** ARGUMENTS: RECORDLENGTH ****
1E6B	17B4	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
1E6B	17B4	REM **** TRANSMITTED ****
1E6B	17B4	REM *****
1E6B	17B4	9020 REM
1E6C	17B4	ERRSTAT\$ = SPACE\$(60)
1E78	17B4	RECORDPTR = VARPTR(SHEETREC(0))
1E7F	17B4	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
1E98	17B4	ERRMSG\$=""
1EA1	17B4	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
1EC2	17B4	GOTO 9100
1EC6	17B4	
1EC6	17B4	REM *****
1EC6	17B4	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
1EC6	17B4	REM ****
1EC6	17B4	REM **** ARGUMENTS: PRINTPOS ****
1EC6	17B4	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
1EC6	17B4	REM **** VALUES = 0 THRU 90 ****
1EC6	17B4	REM ****
1EC6	17B4	REM **** ARGUMENTS: PSTRINGS ****
1EC6	17B4	REM **** TEXT TO BE PRINTED ON THE FORM ****
1EC6	17B4	REM ****
1EC6	17B4	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
1EC6	17B4	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
1EC6	17B4	REM *****
1EC6	17B4	9030 REM

RACP830

EMERGENCY ROOM FORM DESTROY/DECODE PROGRAM

PAGE 26

07-06-87

14:57:17

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
1EC7	17B4	ERRSTAT\$ = SPACES(60)
1ED3	17B4	RECORDPTR = VARPTR(SHEETREC(0))
1EDA	17B4	CALL TPRINT(PRINTPOS,PSTRINGS,ERRSTAT\$)
1EEF	17BA	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
1F0B	17BA	GOTO 9100
1F0F	17BA	
1F0F	17BA	9100 RETURN
1F12	17BA	REM -----END OF SUBROUTINE RACS9000.SUB -----
1F12	17BA	
1F12	17BA	REM END OF SUBROUTINES =====
1F12	17BA	
1F12	17BA	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
1F13	17BA	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1F1B	17BA	LPRINT "ERASING FILE ";DATFIL\$
1F28	17BA	BEEP
1F2C	17BA	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1F38	17BA	CLOSE
1F3C	17BA	OPEN DATFIL\$ FOR OUTPUT AS #1
1F4E	17BA	PRINT #1,STRING\$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
1F60	17BA	CLOSE
1F64	17BA	
1F64	17BA	30000 REM
1F65	17BA	CLOSE
1F69	17BA	CHAIN "RACP10"
1F70	17BA	
1F73	17BA	

22151 Bytes Available

15513 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

PAGE 1

07-06-87

14:47:41

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

001A 0002 REM \$LINESIZE: 132

001A 0002 REM \$PAGESIZE: 66

001A 0002 REM \$TITLE: 'RACP850 '

001A 0002 REM \$SUBTITLE: 'GROUP FORM I & II DESTSTRING/DECODE PROGRAM'

001A 0002 REM \$PAGE

GROUP FORM I & II DESTING/DECODE PROGRAM

05-06-87

\$1.00

IBM Personal Computer BASIC Compiler V1.00

```

Offset  Data      Source Line
001A 0002 REM +-----+
001A 0002 REM | NAME: RACP850          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | DATE: 10 JUN 87      GROUP FORM I & II PROGRAM          |
001A 0002 REM | D R BOLLING                                     |
001A 0002 REM +-----+
001A 0002 REM          GROUP FORM I & II OMR INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the base form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RACP05, otherwise,
001A 0002 REM the program chains to program RACP10 on exit.
001A 0002 REM
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 REM *****
001A 0002 REM * NAME: RACDIM.MOD          DIMENSION DEFINITIONS *
001A 0002 REM * Date: 28 Feb 84          Written by: Floyd Cole *
001A 0002 REM *****
001A 0002 REM ' Dimensioned variables are defined in this file.
001A 0002 REM ' It is an included file so it cannot be run in a stand-alone,
001A 0002 REM ' mode.
001A 0002 REM '
001A 0002 REM ' This program segment may be modified, but all files containing
001A 0002 REM ' an include for this segment must be re-compiled in order to
001A 0002 REM ' affect the changes made here.
001A 0002 REM ' ***** START OF DIMENSION DEFINITION *****
001A 0002 REM
001A 0002 REM DEFINIT A-Z
001A 0002 REM DIM USER$(2),MOLENGTH(12),DATEERR$(3)
001A 0002 REM
001A 0002 REM ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002 REM
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002 REM
001A 0002 REM DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 REM DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 REM DIM YN$(3) '(YES/NO ANSWERS 0=?, 1 = Y, 2=N, 3=X)
001A 0002 REM DIM ED.MSG$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002 REM DIM CLINIC1.PFX$(5) '(PREFIX -B D F G S- FOR CLINIC #1)
001A 0002 REM DIM CLINIC2.PFX$(6) '(PREFIX -A B C D F S- FOR CLINIC #2)
001A 0002 REM DIM PROCED$(098) '(PROCEDURE TABLE FOR BASE FORM)
001A 0002 REM DIM HOLD$(99) '(HOLD AREA FOR SUBROUTINE 6000)
001A 0002 REM DIM SPECL$(09) '(SPECIAL PROGRAMS)

```

Offset Data Source Line

```

001A 0002
001A 0002 REM $INCLUDE: 'RACCMN.MOD'      REM INCLUDE THE COMMON AREA DEFINITION
001A 0002 *****
001A 0002 '* NAME: RACCMN.MOD              COMMON AREA DEFINITION *
001A 0002 '* Date: 28 Feb 84              Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A 0002 ' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re*compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 '
001A 0002 ' *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002 COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL$ 'BASIC SCREEN COLORS
001A 0002 COMMON HEADERS$ '21 CHARACTER SCANNER HEADER INFO
001A 0002 COMMON TEXT$ '11 AINING CHARACTERS FROM SCANNER
001A 0002 COMMON PGMIDS$ 'PROGRAM OR FORM ID
001A 0002 COMMON MOLENGTH( ) 'DAYS IN THE MONTH
001A 0002 COMMON USERS( )
001A 0002 ' *****END OF COMMON DEFINITION*****
001A 0002
001A 0002 REM $INCLUDE: 'RACDEF.MOD'      REM INCLUDE THE DEFAULT DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACP01.DEF              DEFAULT DEFINITIONS *
001A 0002 '* Date: 28 Feb 84              Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' Variables used in common that have a default value on start*up
001A 0002 ' will be held in this file. It is an included file so it cannot
001A 0002 ' be run in a stand*alone mode. In normal operation, this file
001A 0002 ' should be 'included' in the main program only (RACP10.BAS).
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re*compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 '
001A 0002 ' *****START OF DEFAULT DEFINITION*****
001A 0002
001A 0002 FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
0046 1104 BACK = 1 'Background Color = Light Blue
004D 1104 BORD = 4 'BORDER = RED
0054 1106 HIDE = 4 'ALTERNATE COLOR = RED
0058 1106 EFORE= 14 'ERROR FOREGROUND DISPLAY
0062 1106 EBACK= 0 'ERROR BACKGROUND DISPLAY
0069 1106 BELL$ = CHR$(7) 'Sound the bell
0075 1106
0075 1106 MOLENGTH(1) = 31 'JAN
007C 1106 MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
0083 1106 MOLENGTH(3) = 31 'MAR
008A 1106 MOLENGTH(4) = 30 'APR
0091 1106 MOLENGTH(5) = 31 'MAY
0098 1106 MOLENGTH(6) = 30 'JUN

```

Offset Data Source Line

```

009F 11D6      MOLENGTH(7)  = 31      'JUL
00A6 11D6      MOLENGTH(8)  = 31      'AUG
00AD 11D6      MOLENGTH(9)  = 30      'SEP
00B4 11D6      MOLENGTH(10) = 31      'OCT
00B8 11D6      MOLENGTH(11) = 30      'NOV
00C2 11D6      MOLENGTH(12) = 31      'DEC
00C9 11D6
00C9 11D6      DATEERR$(0)  = " "
00D2 11D6      DATEERR$(1)  = "INVALID MONTH"
00D8 11D6      DATEERR$(2)  = "INVALID DAY "
00E4 11D6      DATEERR$(3)  = "DAY TOO LARGE FOR MONTH CODED"
00ED 11D6
00ED 11D6      MAXLENGTH    = 80      'MAXIMUM LENGTH OF OUTPUT RECORD
00F4 11D8      PAD$         = " "      'PAD CHARACTER FOR SHORT RECORDS
00FD 11DC
00FD 11DC      *****END OF DEFAULT DEFINITION*****
00FD 11DC
00FD 11DC
00FD 11DC      KEY OFF
0103 11DC
0103 11DC      REM *****
0103 11DC
0103 11DC      REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0103 11DC      REM      CLINIC #1          CLINIC #2
0103 11DC      CLINIC1.PFX$(0)=" "      : CLINIC2.PFX$(0)=" "
0115 11DC      CLINIC1.PFX$(1)="B"      : CLINIC2.PFX$(1)="A"
0127 11DC      CLINIC1.PFX$(2)="D"      : CLINIC2.PFX$(2)="B"
0139 11DC      CLINIC1.PFX$(3)="F"      : CLINIC2.PFX$(3)="C"
0148 11DC      CLINIC1.PFX$(4)="G"      : CLINIC2.PFX$(4)="D"
015D 11DC      CLINIC1.PFX$(5)="S"      : CLINIC2.PFX$(5)="F"
016F 11DC      CLINIC2.PFX$(6)="S"
0178 11DC
0178 11DC      REM *****
0178 11DC      REM PROCEDURE TABLE
0178 11DC      PROCED$(00)=" " : PROCED$(33)="07127" : PROCED$(66)="06055"
0193 11DC      PROCED$(01)="02008" : PROCED$(34)="07128" : PROCED$(67)="03101"
01AE 11DC      PROCED$(02)="02015" : PROCED$(35)="02600" : PROCED$(68)="04240"
01C9 11DC      PROCED$(03)="02080" : PROCED$(36)="07129" : PROCED$(69)="04128"
01E4 11DC      PROCED$(04)="02081" : PROCED$(37)="07130" : PROCED$(70)="04241"
01FF 11DC      PROCED$(05)="02082" : PROCED$(38)="07131" : PROCED$(71)="04242"
021A 11DC      PROCED$(06)="07100" : PROCED$(39)="02596" : PROCED$(72)="04137"
0235 11DC      PROCED$(07)="07101" : PROCED$(40)="02597" : PROCED$(73)="04126"
0250 11DC      PROCED$(08)="07102" : PROCED$(41)="02598" : PROCED$(74)="06054"
0268 11DC      PROCED$(09)="07103" : PROCED$(42)="02599" : PROCED$(75)="06088"
0286 11DC      PROCED$(10)="07104" : PROCED$(43)="02600" : PROCED$(76)="06052"
02A1 11DC      PROCED$(11)="07105" : PROCED$(44)="02601" : PROCED$(77)="06056"
02BC 11DC      PROCED$(12)="07106" : PROCED$(45)="02100" : PROCED$(78)="06089"
02D7 11DC      PROCED$(13)="07107" : PROCED$(46)="02101" : PROCED$(79)="06057"
02F2 11DC      PROCED$(14)="07108" : PROCED$(47)="02102" : PROCED$(80)="06090"
0300 11DC      PROCED$(15)="07109" : PROCED$(48)="02103" : PROCED$(81)="06035"
0328 11DC      PROCED$(16)="07110" : PROCED$(49)="02104" : PROCED$(82)="06055"
0343 11DC      PROCED$(17)="07111" : PROCED$(50)="02600" : PROCED$(83)="06079"
035E 11DC      PROCED$(18)="07112" : PROCED$(51)="03050" : PROCED$(84)="02590"
0379 11DC      PROCED$(19)="07113" : PROCED$(52)="03051" : PROCED$(85)="02591"

```

Offset	Data	Source Line
0394	11DC	PROCD\$(20)="07114" : PROCD\$(53)="03052" : PROCD\$(86)="06056"
03AF	11DC	PROCD\$(21)="07115" : PROCD\$(54)="03053" : PROCD\$(87)="02592"
03CA	11DC	PROCD\$(22)="07116" : PROCD\$(55)="03054" : PROCD\$(88)="02593"
03E5	11DC	PROCD\$(23)="07117" : PROCD\$(56)="03055" : PROCD\$(89)="04137"
0400	11DC	PROCD\$(24)="07118" : PROCD\$(57)="03056" : PROCD\$(90)="02594"
0418	11DC	PROCD\$(25)="07119" : PROCD\$(58)="03057" : PROCD\$(91)="02450"
0436	11DC	PROCD\$(26)="07120" : PROCD\$(59)="03058" : PROCD\$(92)="02455"
0451	11DC	PROCD\$(27)="07121" : PROCD\$(60)="03059" : PROCD\$(93)="02456"
046C	11DC	PROCD\$(28)="07122" : PROCD\$(61)="03060" : PROCD\$(94)="02457"
0487	11DC	PROCD\$(29)="07123" : PROCD\$(62)="03061" : PROCD\$(95)="04137"
04A2	11DC	PROCD\$(30)="07124" : PROCD\$(63)="03062" : PROCD\$(96)="02458"
04BD	11DC	PROCD\$(31)="07125" : PROCD\$(64)="03063" : PROCD\$(97)="02459"
04D8	11DC	PROCD\$(32)="07126" : PROCD\$(65)="03100" : PROCD\$(98)="90768"
04F3	11DC	
04F3	11DC	REM *****
04F3	11DC	REM YES/NO TABLE
04F3	11DC	YNS(0)=" " : YNS(1)="Y" : YNS(2)="N" : YNS(3)="X"
0517	11DC	
0517	11DC	
0517	11DC	REM *****
0517	11DC	PNUM=VAL(PGMID\$)/10
0529	11DE	REM LENGTH OF STRING RECEIVED FROM THE OMR....
0529	11DE	HEADER = 21
0530	11E0	RESPONSE= 173
0537	11E2	RECORDLENGTH = HEADER + RESPONSE
0542	11E4	
0542	11E4	N.PROC = 98 ' NUMBER OF PROCEDURES FOR THIS FORM
0549	11E6	
0549	11E6	DATFIL\$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
0552	11EA	BTIME\$=TIME\$ 'SCAN START TIME
0558	11EE	
0558	11EE	REM *** ENCOUNTER FORM PROCEDURE TABLE ***
0558	11EE	
0558	11EE	REM INCLUDE: 'UCACAMP.OPT' INCLUDE THE OUTP UCA VALIDATE TABLE
0558	11EE	REM INCLUDE: 'UCACAMP.IPT' INCLUDE THE INP UCA VALIDATE TABLE
0558	11EE	
0558	11EE	REM \$PAGE

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

0558 11EE      GOSUB 1000      'MAKE SURE THEY ARE LOGGED ON
0560 11EE      CLS
0564 11EE      GOSUB 7000      'PRINT SCREEN HEADING
0569 11EE
0569 11EE  REM *****
0569 11EE  REM ****          OPEN FILE TO CONTAIN SCANNED DATA          ****
0569 11EE  REM *****
0569 11EE  REM
0569 11EE      OPEN DATFIL$ FOR APPEND AS #1
0578 11EE
0578 11EE  REM *****
0578 11EE  REM ****          CLEAR AND DISPLAY PROGRAM SCREEN          ****
0578 11EE  REM *****
0578 11EE      LPRINT CHR$(15);
0586 11EE      WIDTH "LPT1:",160
0590 11EE      PAGE = 0 : GOSUB 7100      'LINE PRINTER HEADING
059C 11F0      COLOR 14
05A3 11F0      LOCATE 11,26 : PRINT "GROUP I & II FORMS "
05B8 11F0      COLOR FORE,BACK,BORD
05CE 11F0
05CE 11F0  REM *****
05CE 11F0  REM ****          COMMUNICATIONS SETUP          ****
05CE 11F0  REM *****
05CE 11F0  REM PROTOCOL
05CE 11F0      GOSUB 9001
05D3 11F0      IF ERRMSG$ > " " THEN LPRINT ERRMSG$ : GOTO 30000
05ED 11F4
05ED 11F4  REM START SCANNER (SI)
05ED 11F4      CNTRLPT = 1 : GOSUB 9010
05F9 11F6      IF ERRMSG$ > " " THEN LPRINT ERRMSG$ : GOTO 30000
0613 11F6
0613 11F6      LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING  "
0628 11F6      READTYPE=3          'FIRST TIME IN.. SCANNER IS STARTED..
062F 11F8
062F 11F8  REM *****
062F 11F8  REM ***          SET SCAN SHEET CALL          ***
062F 11F8  REM *****
062F 11F8
062F 11F8  REM *****
062F 11F8  REM ***          BEGIN WITH READING GROUP FORM I          ***
062F 11F8  REM *****
062F 11F8
062F 11F8  6      AS=INKEY$
0638 11FC      IF AS=CHR$(27) THEN GOTO 25000
064E 11FC
064E 11FC
064E 11FC      GOSUB 9020          'SCAN SUBROUTINE - GET A RECORD
0653 11FC      IF MID$(ERRSTAT$,14,3)="415" THEN GOTO 25000
066F 1200      READTYPE=2      'CHANGE TO READ SUBSEQUENT SHEETS
0676 1200      TEXT$=""      'CLEAR THE INPUT AREA
067F 1200      TEXTII$=""      'CLEAR THE INPUT AREA FOR GROUP FORM II
0688 1204      GOSUB 8000      'DECODE HEADER
068D 1204      GOSUB 8050      'CHECK FOR END OF JOB/END OF BATCH
0692 1204      GOSUB 8200      'DECODE THE RESPONSE POSITIONS

```

Offset	Data	Source Line
0697	1204	
0697	1204	7 LITHOS = MID\$(TEXT\$,22,8)
06A9	1208	GOSUB 8070 'CHECK FOR SCANNER ERRORS
06AE	1208	GOSUB 8100 'PRINT THE DATA ON THE SCREEN
0683	1208	
0683	1208	REM *** EXPECT A GROUP FORM I TO BE READ HERE *****
0683	1208	REM *** CHARACTERS IN POSITION 22 AND 23 SHOULD BE NONBLANK
0683	1208	REM *** IF TYPE I *****
0683	1208	IF MID\$(TEXT\$,22,2)<>" " THEN 8 'GO IF OK
06C8	1208	N.ERR=1
06D2	120A	ED.MSG\$(N.ERR)="GROUP FORM I NOT READ-CHECK SEQUENCE"
06E6	120A	GOSUB 995 'PRINT ERROR AND REJECT
06EB	120A	GOTO 6 'GO BACK AND READ NEXT SHEET
06EF	120A	
06EF	120A	8 GOSUB 100 'EDIT GROUP I FORM RECORD
06F4	120A	
06F4	120A	REM *** IS THERE A CONTINUATION ? *****
06F4	120A	IF CSHEETS="1" THEN 12 'READ NEXT FORM II - CONTINUATION
0702	120E	
0702	120E	GOSUB 200 'WRITE THE OUTPUT
0707	120E	GOTO 6 'BACK TO READ NEXT SHEET
0708	120E	
0708	120E	REM *****
0708	120E	REM *** READ GROUP FORM II ***
0708	120E	REM *****
0708	120E	12 REM
070C	120E	
070C	120E	EOJ1=0
0713	1210	AS=INKEY\$
071C	1210	IF AS=CHR\$(27) THEN GOTO 25000
0732	1210	
0732	1210	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
0737	1210	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
0753	1210	TEXT\$="" 'CLEAR THE INPUT AREA
075C	1210	GOSUB 8000 'DECODE HEADER
0761	1210	
0761	1210	REM *** EOF TEST *****
0761	1210	IF PROGRAM\$ = PGMID\$ THEN 14 'GO ON IF NOT EOF
076F	1214	
076F	1214	N.ERR=N.ERR + 1
0777	1214	ED.MSG\$(N.ERR)="EOF WHEN EXPECTING A GROUP II FORM"
0788	1214	GOSUB 995 'PRINT ERROR AND REJECT
0790	1214	GOSUB 8050 'GO QUIT
0795	1214	
0795	1214	14 GOSUB 8200 'DECODE THE RESPONSE POSITIONS
079A	1214	GOSUB 8070 'CHECK FOR SCANNER ERRORS
079F	1214	
079F	1214	REM *** IS THIS FORM II - IT SHOULD BE *****
079F	1214	IF MID\$(TEXT\$,22,2)=" " THEN 16 'GO CHECK LITHO MATCH
07B7	1214	
07B7	1214	N.ERR=N.ERR+1
07BF	1214	ED.MSG\$(N.ERR)="NO GROUP II FORMS READ FOR THIS LITHO"
07D3	1214	LPRINT "LITHO # ";LITHOS;" ...ERRORS"
07E5	1214	FOR I=1 TO N.ERR

Offset	Data	Source Line
07F2	1216	LPRINT USING "### ";1997;
07FE	1218	LPRINT "=> ";ED.MSG\$(1997)
0814	1218	NEXT 1997
0825	1218	GOTO 7 'FORGET THE GROUP II AND EDIT THIS ONE
0829	1218	
0829	1218	REM *** CHECK TO SEE IF LITHO MATCHES *****
0829	1218	16 IF MID\$(TEXT\$,24,6)=MID\$(LITHO\$,3,6) THEN 18 'OK IF MATCH
0848	1218	
0848	1218	N.ERR=N.ERR+1
0850	1218	ED.MSG\$(N.ERR)="LITHO MISMATCH FROM FORM I TO FORM II"
0864	1218	GOSUB 995 'PRINT ERROR AND REJECT
0869	1218	GOTO 6 'START OVER
0860	1218	
0860	1218	18 REM *** ACCUMULATE THE GROUP FORM II INFO
086E	1218	TEXTII\$=TEXTII\$+MID\$(TEXT\$,30,165)
0883	1218	
0883	1218	REM *****
0883	1218	REM *** READ NEXT RECORD - UNSURE IF I OR II OR EOF ***
0883	1218	REM *****
0883	1218	20 AS=INKEY\$
088C	1218	IF AS=CHR\$(27) THEN GOTO 25000
08A2	1218	
08A2	1218	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
08A7	1218	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
08C3	1218	
08C3	1218	TEXT\$="" 'CLEAR THE INPUT AREA
08CC	1218	GOSUB 8000 'DECODE HEADER
08D1	1218	
08D1	1218	REM *** EOF TEST *****
08D1	1218	IF PROGRAM\$ <> PGMIID\$ THEN 30 'TIME TO QUIT
08DF	1218	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
08E4	1218	
08E4	1218	REM *** IS THIS FORM I OR II *****
08E4	1218	IF MID\$(TEXT\$,22,2)=" " THEN 28 'GO IF FORM II
08FC	1218	
08FC	1218	REM *** FORM I READ - PROCESS OLD INFO BEFORE PROCEEDING *****
08FC	1218	
08FC	1218	GOSUB 200 'WRITE THE OLD INFO OUT
0901	1218	
0901	1218	TEXTII\$="" 'RESET FORM II INFO BUFFER
090A	1218	LITHO\$ = MID\$(TEXT\$,22,8)
091C	1218	GOSUB 8070 'CHECK FOR SCANNER ERRORS
0921	1218	GOTO 8 'BACK TO EDIT THE NEW FORM I INFO
0925	1218	
0925	1218	28 GOSUB 8070 'CHECK FOR SCANNER ERRORS
092A	1218	GOTO 16 'THEN BACK TO PROCESS THE FORM II SHEET
092E	1218	
092E	1218	30 GOSUB 200 'WRITE THE LAST INFO OUT
0933	1218	GOSUB 8050 'LET THIS SUBROUTINE QUIT FOR US
0938	1218	
0938	1218	REM \$INCLUDE: 'RACH850.MO1' INCLUDE THE READ FORM REFORMAT/EDIT MOD
0938	1218	REM *****
0938	1218	REM **** AMBULATORY CARE INFORMATION SYSTEM 10 JUN 87 ****
0938	1218	REM **** D R BOLLING ****

Offset Date Source Line

IBM Personal Computer BASIC Compiler V1.00

```

0938 1218 REM **** MODULE NAME : RACB50.M01 ****
0938 1218 REM **** SCANNER PROGRAM # : GROUP I FORM ****
0938 1218 REM **** PURPOSE : REFORMAT/EDIT THE FORM ****
0938 1218 REM ****
0938 1218 REM *****
0938 1218 REM **** RESERVED LINE NUMBERS 100-199 ****
0938 1218 REM *****
0938 1218
0938 1218 100 N.ERR =0 'COUNTS THE NUMBER OF ERRORS
093F 1218
093F 1218 REM *** LITHO CODE DONE IN BAS PROGRAM ***
093F 1218 REM *** CLINIC ID (PREFIX + CODE) ***
093F 1218 102 CK.X=VAL(MID$(TEXT$,30,1))
0955 121A CK.CODS=MID$(TEXT$,31,3)
0967 121E REM GOSUB 5500 'OUTP UCA CODE CHECK
0967 121E REM IF RT.5500 = 0 THEN GOTO 102
0967 121E REM N.ERR = N.ERR + 1
0967 121E REM ED.MSG$(N.ERR)="INVALID CLINIC CODE"
0967 121E
0967 121E CL1.CODS=CLINIC1.PFX$(CK.X) + CK.CODS 'HAND CODED
097F 1222
097F 1222 REM VISIT DATE
097F 1222 104 CMS=MID$(DATES,1,2) 'CURRENT MONTH
0991 1226 YRS=MID$(DATES,9,2) 'CURRENT YEAR
09A3 122A XS=MID$(TEXT$,34,4) 'MONTH AND DAY
09B5 122E IF LEFT$(XS,2)<=CMS THEN 105 'OK, USE THIS YEAR
09CA 122E YRS=RIGHT$(STR$(VAL(YRS)-1),2) 'USE LAST YEAR
09EA 122E 105 VDATES=YRS+XS
09F8 1232 D1$=MID$(XS,3,1)
0A0A 1236 D2$=MID$(XS,4,1)
0A1C 123A IF D1$=" " AND D2$<>" " THEN RT.5000=2 : GOTO 106
0A4A 123C IF D1$<>" " AND D2$=" " THEN RT.5000=2 : GOTO 106
0A78 123C 'EDIT VISIT DATE
0A78 123C CK.5000$=VDATES$
0A81 1240 GOSUB 5000 'DATE CHECK
0A86 1240 IF RT.5000=0 THEN GOTO 107
0A95 1240 106 N.ERR=N.ERR+1
0A9D 1240 ED.MSG$(N.ERR)="VISIT DATE" + DATEERR$(RT.5000)
0AC3 1240
0AC3 1240 REM *** PRIMARY PROVIDER ***
0AC3 1240 107 PPROV.PFX$=MID$(TEXT$,38,1)
0AD5 1244 PPROV.NUM$=MID$(TEXT$,39,4)
0AE7 1248 REM *** FORM NUMBER ***
0AE7 1248 FRM$=LEFT$(PGMID$,2)
0AF6 124C
0AF6 124C REM *** VISIT COUNT ***
0AF6 124C 112 VCNT$=MID$(TEXT$,43,1)
0B08 1250 REM *** PRIMARY PROVIDER TIME SPENT ***
0B08 1250 113 PPROV.TIM$=MID$(TEXT$,44,3)
0B1A 1254
0B1A 1254 REM *** SECONDARY PROVIDER ***

```

Offset	Data	Source Line
081A	1254	SPROV.PFX\$=MID\$(TEXT\$,47,1)
082C	1258	SPROV.NUM\$=MID\$(TEXT\$,48,4)
083E	125C	IF SPROV.PFX\$+SPROV.NUM\$ = " " THEN SPBL=1 ELSE SPBL=0
0863	125E	IF SPROV.PFX\$=" " AND SPROV.NUM\$<>" " THEN 114
0889	125E	IF SPROV.PFX\$<>" " AND SPROV.NUM\$=" " THEN 114
08AF	125E	GOTO 115
08B3	125E	
08B3	125E	114 N.ERR = N.ERR + 1
08B8	125E	ED.MSG\$(N.ERR)="PROV 2 CODE MISSING PREFIX OR NUMBER"
08CF	125E	
08CF	125E	
08CF	125E	REM *** SECONDARY PROVIDER TIME ***
08CF	125E	115 TEMP\$=MID\$(TEXT\$,52,3)
08E1	1262	REM PAD WITH LEADING ZEROS
08E1	1262	IF TEMP\$=" " THEN TEMP\$="000"
08F8	1262	IF LEFT\$(TEMP\$,2)=" " THEN TEMP\$="00" + RIGHT\$(TEMP\$,1)
0C22	1262	IF LEFT\$(TEMP\$,1)=" " THEN TEMP\$="0" + RIGHT\$(TEMP\$,2)
0C4C	1262	SPROV.TIM\$=TEMP\$
0C55	1266	REM IS THERE A TIME AND NO SEC PROV CODED?
0C55	1266	IF SPROV.TIM\$ > "000" AND SPBL = 1 THEN 116 'GO IF YES
0C78	1266	REM IS THERE NO TIME AND A SEC PROV CODED?
0C78	1266	IF SPROV.TIM\$ = "000" AND SPBL = 0 THEN 117 'GO IF YES
0C9B	1266	GOTO 118
0C9F	1266	
0C9F	1266	116 N.ERR = N.ERR + 1
0CA7	1266	ED.MSG\$(N.ERR)="TIME CODED WITH NO SEC PROV CODED"
0CBB	1266	GOTO 118
0CBF	1266	
0CBF	1266	117 N.ERR = N.ERR + 1
0CC7	1266	ED.MSG\$(N.ERR)="NO PROV 2 TIME"
0CDB	1266	
0CDB	1266	118 REM
0CDC	1266	REM *** UNIT ID CODE ***
0CDC	1266	120 UIC\$=MID\$(TEXT\$,55,6)
0CEE	126A	
0CEE	126A	REM *** TIME PREP PROV 1 ***
0CEE	126A	122 TIMPR1\$=MID\$(TEXT\$,61,3)
0D00	126E	
0D00	126E	REM *** TIME PREP PROV 2 ***
0D00	126E	124 TIMPR2\$=MID\$(TEXT\$,64,3)
0D12	1272	
0D12	1272	REM *** TIME TRAV PROV 1 ***
0D12	1272	126 TIMTR1\$=MID\$(TEXT\$,67,3)
0D24	1276	
0D24	1276	REM *** TIME TRAV PROV 2 ***
0D24	1276	128 TIMTR2\$=MID\$(TEXT\$,70,3)
0D36	127A	
0D36	127A	REM *** NO. ARMY ACTIVE ***
0D36	127A	130 NACT\$=MID\$(TEXT\$,73,3)
0D48	127E	
0D48	127E	REM *** NO. OTHER ACTIVE ***
0D48	127E	132 NOTH\$=MID\$(TEXT\$,76,3)
0D5A	1282	
0D5A	1282	REM *** NO. RETIRED ***

Offset	Data	Source Line
005A	1282	134 NRETS=MID\$(TEXT\$,79,3)
006C	1286	
006C	1286	REM *** NO. DEPENDENTS ***
006C	1286	136 NDEPS=MID\$(TEXT\$,82,3)
007E	128A	
007E	128A	REM *** NO. CIVILIANS ***
007E	128A	138 NCIVS=MID\$(TEXT\$,85,3)
0090	128E	
0090	128E	REM *** CONT. SHEET ***
0090	128E	140 CSHEETS=MID\$(TEXT\$,88,1)
00A2	128E	
00A2	128E	REM *** ADDITIONAL PROCEDURE 1 ***
00A2	128E	146 ADDP1\$=MID\$(TEXT\$,89,5)
00B4	1292	
00B4	1292	REM *** EVALUATION/SERV/PROC PROV 1 ***
00B4	1292	150 X=VAL(MID\$(TEXT\$,94,2))
00CA	1294	PROCS=PROCD\$(X)
00DC	1298	
00DC	1298	IF ADDP1\$=" " AND PROCS=" " THEN 152
00FF	1298	IF ADDP1\$<>" " AND PROCS<>" " THEN 154
0E22	1298	IF PROCS=" " THEN PROCS=ADDP1\$
0E39	1298	GOTO 199
0E3D	1298	
0E3D	1298	152 N.ERR = N.ERR + 1
0E45	1298	ED.MSG\$(N.ERR)="NO REASON CODED"
0E59	1298	GOTO 199
0E5D	1298	
0E5D	1298	154 N.ERR = N.ERR + 1
0E65	1298	ED.MSG\$(N.ERR)="MORE THAN ONE REASON CODED"
0E79	1298	
0E79	1298	199 REM
0E7A	1298	RETURN
0E7D	1298	REM -----END OF MODULE RACM850.M01-----
0E7D	1298	REM \$INCLUDE: 'RACM850.M02' REM INCLUDE THE BASE FORM DISK WRITER
0E7D	1298	REM *****
0E7D	1298	REM **** AMBULATORY CARE INFORMATION SYSTEM 11 JUN 87 ****
0E7D	1298	REM **** D R BOLLING ****
0E7D	1298	REM **** MODULE NAME : RACM850.MOD ****
0E7D	1298	REM **** SCANNER PROGRAM # : GROUP FORM I & II ****
0E7D	1298	REM ****
0E7D	1298	REM **** PURPOSE : CREATE AND WRITE THE DISK ****
0E7D	1298	REM **** RECORD FOR INPUT TO FOCUS ****
0E7D	1298	REM ****
0E7D	1298	REM *****
0E7D	1298	REM **** RESERVED LINE NUMBERS 200-299 ****
0E7D	1298	REM *****
0E7D	1298	
0E7D	1298	REM BUILD THE OUTPUT RECORD
0E7D	1298	
0E7D	1298	200 GOSUB 276 'BUILD THE RECORD KEY
0E82	1298	
0E82	1298	REM *****
0E82	1298	REM **** RECORD TYPE "1" - MAIN TRANSACTION ****
0E82	1298	REM *****

Offset	Data	Source Line
0EB2	1298	
0EB2	1298	GOSUB 278 'BUILD DATA FOR TYPE 1
0EB7	1298	RECOU\$ =PGMID\$+"1"+RECKEY\$+RECOD1\$ 'TRANSACTION ID PLUS RECORD
0EA1	12A4	GOSUB 280
0EA6	12A4	PRINT #1,RECOU\$
0EB1	12A4	
0EB1	12A4	REM *****
0EB1	12A4	REM **** RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE ****
0EB1	12A4	REM *****
0EB1	12A4	
0EB1	12A4	202 RECOD2\$=PROC\$
0EBA	12A8	
0EBA	12A8	RECOU\$=PGMID\$+"2"+RECKEY\$+"1"+RECOD2\$ 'TRANSACTION ID PLUS RECORD
0EDA	12A8	GOSUB 280
0EDF	12A8	PRINT #1,RECOU\$
0EEA	12A8	
0EEA	12A8	204 REM END OF TYPE 2 RECORDS
0EEB	12A8	REM *****
0EEB	12A8	REM **** RECORD TYPE "3" - RECKEY PLUS SPECIFIC PRE CLINIC CODES ****
0EEB	12A8	REM *****
0EEB	12A8	220 REM END OF TYPE 3 RECORDS
0EEC	12A8	
0EEC	12A8	REM *****
0EEC	12A8	REM **** RECORD TYPE "4" - RECKEY PLUS OTHER CODES ****
0EEC	12A8	REM *****
0EEC	12A8	230 REM END OF TYPE 4 RECORDS
0EED	12A8	
0EED	12A8	REM *****
0EED	12A8	REM **** RECORD TYPE "5" - RECKEY PLUS GROUP I DATA ****
0EED	12A8	REM *****
0EED	12A8	232 REM
0EEE	12A8	
0EEE	12A8	REM *** UNIT ID CODE ***
0EEE	12A8	RECOD5\$= UIC\$
0EF7	12AC	
0EF7	12AC	REM *** TIME PREP PROV 1 ***
0EF7	12AC	RECOD5\$ = RECOD5\$ + TIMPR1\$
0F03	12AC	
0F03	12AC	REM *** TIME PREP PROV 2 ***
0F03	12AC	RECOD5\$ = RECOD5\$ + TIMPR2\$
0F0F	12AC	
0F0F	12AC	REM *** TIME TRAV PROV 1 ***
0F0F	12AC	RECOD5\$ = RECOD5\$ + TIMTR1\$
0F18	12AC	
0F18	12AC	REM *** TIME TRAV PROV 2 ***
0F18	12AC	RECOD5\$ = RECOD5\$ + TIMTR2\$
0F27	12AC	
0F27	12AC	REM *** NO. ARMY ACTIVE ***
0F27	12AC	RECOD5\$ = RECOD5\$ + NACT\$
0F33	12AC	
0F33	12AC	REM *** NO. OTHER ACTIVE ***
0F33	12AC	RECOD5\$ = RECOD5\$ + NOTH\$
0F3F	12AC	
0F3F	12AC	REM *** NO. RET MILITARY ***

Offset	Data	Source Line
0F3F	12AC	RECOD5\$ = RECOD5\$ + NRET\$
0F4B	12AC	
0F4B	12AC	REM *** NO. DEPENDENTS ***
0F4B	12AC	RECOD5\$ = RECOD5\$ + NDEP\$
0F57	12AC	
0F57	12AC	REM *** NO. CIVILIANS ***
0F57	12AC	RECOD5\$ = RECOD5\$ + NCIV\$
0F63	12AC	
0F63	12AC	REM *** CONTINUATION SHEET ***
0F63	12AC	RECOD5\$ = RECOD5\$ + CSHEETS
0F6F	12AC	
0F6F	12AC	RECOU\$=PGMID\$+"5"+RECKEY\$+RECOD5\$ 'TRANSACTION ID PLUS RECORD
0F89	12AC	GOSUB 280
0F8E	12AC	PRINT #1,RECOU\$
0F99	12AC	
0F99	12AC	240 REM END OF TYPE 5 RECORDS
0F9A	12AC	
0F9A	12AC	REM *****
0F9A	12AC	REM **** RECORD TYPE "A" - RECKEY PLUS SSN AND FMP ****
0F9A	12AC	REM *****
0F9A	12AC	
0F9A	12AC	PTR=1
0FA1	12AE	242 IF LEN(TEXT11\$)< PTR THEN 250
0FB0	12AE	RECODA\$=MID\$(TEXT11\$,PTR,11)
0FC3	12B2	IF RECODA\$=SPACE\$(11) THEN 244
0FD5	12B2	
0FD5	12B2	RECOU\$=PGMID\$+"A"+RECKEY\$+RECODA\$ 'TRANSACTION ID PLUS RECORD
0FEF	12B2	GOSUB 280
0FF4	12B2	PRINT #1,RECOU\$
0FFF	12B2	
0FFF	12B2	244 PTR=PTR+11
1009	12B2	GOTO 242
100D	12B2	
100D	12B2	250 REM END OF TYPE A RECORDS
100E	12B2	GOTO 299
1012	12B2	
1012	12B2	REM *****
1012	12B2	REM **** SUBROUTINE 270 - BUILD THE RECORD KEY ****
1012	12B2	REM *****
1012	12B2	276 RECKEY\$=""
101B	12B2	
101B	12B2	REM *** CLINIC ID (PREFIX + COD) ***
101B	12B2	RECKEY\$= CL1.COD\$
1024	12B2	
1024	12B2	REM *** VISIT DATE ***
1024	12B2	RECKEY\$=RECKEY\$+ VDATE\$
1030	12B2	
1030	12B2	REM *** PRIMARY PROVIDER ***
1030	12B2	RECKEY\$ = RECKEY\$ + PPROV.PFX\$ + PPROV.NUM\$
1043	12B2	
1043	12B2	REM *** PATIENT SSN ***
1043	12B2	RECKEY\$ = RECKEY\$ + STRING\$(9," ")
1055	12B2	
1055	12B2	REM *** FAMILY MEMBER PREF ***

Offset	Data	Source Line
1055	1282	RECKEY\$ = RECKEY\$ + " "
1061	1282	
1061	1282	REM *** LITHO CODE ***
1061	1282	RECKEY\$ = RECKEY\$ + LITHOS
1060	1282	
1060	1282	REM *** FORM NUMBER ***
1060	1282	RECKEY\$ = RECKEY\$ + FRMNS
1079	1282	
1079	1282	RETURN
107C	1282	
107C	1282	REM *****
107C	1282	REM **** SUBROUTINE 272 - BUILD THE DATA FOR TYPE 1 ****
107C	1282	REM *****
107C	1282	278 RECOD1\$=""
1085	1282	
1085	1282	REM *** VISIT COUNT ***
1085	1282	RECOD1\$ = RECOD1\$ + VCNT\$
1091	1282	
1091	1282	REM *** PRIMARY PROV TIME ***
1091	1282	RECOD1\$ = RECOD1\$ + PPROV.TIM\$
1090	1282	
1090	1282	REM *** SECONDARY PROVIDER ***
1090	1282	RECOD1\$ = RECOD1\$ + SPROV.PFX\$ + SPROV.NUM\$
1080	1282	
1080	1282	REM *** SECONDARY PROVIDER TIME ***
1080	1282	RECOD1\$ = RECOD1\$ + SPROV.TIM\$
108C	1282	
108C	1282	REM *** REASON FOR SECONDARY PROVIDER ***
108C	1282	REM *** APPOINTMENT STATUS ***
108C	1282	REM *** REFERRAL CODE ***
108C	1282	REM *** PLACE OF VISIT ***
108C	1282	REM *** JOB RELATED VISIT ***
108C	1282	REM *** MIL ONLY DUTY ***
108C	1282	REM *** MIL ONLY QTRS ***
108C	1282	REM *** MIL ONLY PROFILE ***
108C	1282	REM *** NOT AVAILABLE ***
108C	1282	REM *** ADMITTED ***
108C	1282	REM *** INFIELD ***
108C	1282	REM *** INJURY ***
108C	1282	REM *** PURPOSE OF VISIT ***
108C	1282	REM *** PRIM FOLLOW-UP/RULE OUT ***
108C	1282	REM *** PRIMARY DX CODE ***
108C	1282	RECOD1\$ = RECOD1\$ + STRING\$(22, " ")
10CE	1282	
10CE	1282	RETURN
10D1	1282	
10D1	1282	REM *****
10D1	1282	REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
10D1	1282	REM *****
10D1	1282	280 PAD=MAXLENGTH - LEN(RECOUT\$) 'FIND OUT HOW SHORT THE RECORD IS
100F	1284	RECOUT\$ = RECOUT\$ + STRING\$(PAD,PAD\$) 'PAD THE RECORD WITH FILL CHAR
10F2	1284	RETURN
10F5	1284	
10F5	1284	299 REM

```

Offset  Data   Source Line
10F6 12B4 RETURN
10F9 12B4 REM -----END OF MODULE RACP850.MOD-----
10F9 12B4
10F9 12B4 REM *****
10F9 12B4 REM ***          ERROR CHECK AND PRINT ROUTINE          ***
10F9 12B4 REM *****
10F9 12B4 995 IF N.ERR = 0 THEN GOTO 996
1108 12B4      LPRINT "LITHO # ";LITHO$;" ...ERRORS"
111A 12B4      FOR 1997 = 1 TO N.ERR
1127 12B6          LPRINT USING "### ";1997;
1133 12B6          LPRINT "=> ";ED.MSG$(1997)
1149 12B6      NEXT 1997
115A 12B6      LN.COUNT = LN.COUNT + N.ERR + 1
1166 12B8      CNTRLPT = 6
116D 12B8      GOSUB 9010      'REJECT THE FORM
1172 12B8 996 RETURN
1175 12B8
1175 12B8 998 REM CONTINUE
1176 12B8
1176 12B8 999 GOTO 6
117A 12B8
117A 12B8 REM END OF SCAN/DECODE/WRITE LOOP =====
117A 12B8
117A 12B8 1000 REM $INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
117B 12B8 REM *****
117B 12B8 REM * NAME: RACS1000          LOGON VERIFICATION SUBROUTINE      *
117B 12B8 REM * Date: 28 Feb 84      PATIENT REGISTRATION PROGRAM        *
117B 12B8 REM *****
117B 12B8 REM          PATIENT OMR INPUT PROGRAM                          *
117B 12B8 REM                                                                 *
117B 12B8 REM This program verifies user is logged on properly. If there is no *
117B 12B8 REM valid user logged on at the time of execution, this subroutine will*
117B 12B8 REM chain to the logon program RACP05, otherwise a return is issued.  *
117B 12B8 REM *****
117B 12B8 REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
117B 12B8 REM *****
117B 12B8 1001 OPEN "I",1,"RACLOG.DAT"
118D 12B8      IF EOF(1) THEN 1002          'MAKE THEM LOG ON FIRST
1198 12B8      INPUT #1,USERS$(1),DT$,TMS$,PIDS
11BC 12C4      IF USERS$(1) = "" THEN 1002      'MAKE THEM LOG ON FIRST
11CA 12C4      IF USERS$(1) = "*****" THEN 1002      'MAKE THEM LOG ON FIRST
11D8 12C4      CLOSE 1
11DF 12C4      SCREEN 0,1,0,0
11F5 12C4      COLOR FORE,BACK,BORD
1208 12C4      CLS
120F 12C4      RETURN
1212 12C4
1212 12C4 1002 CLOSE
1216 12C4      CHAIN "RACP05"
121D 12C4      '=====END OF LOGON VERIFY SUBROUTINE 1000=====
121D 12C4
121D 12C4 2000 REM $INCLUDE: 'RACP2000.SUB' INCLUDE THE RIPLY/DELAY SUB
121D 12C4 REM *****

```

```

Offset Data Source Line
121E 12C4 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
121E 12C4 REM **** SKIP COLE ****
121E 12C4 REM **** SUBROUTINE NAME : RACS2000.SUB ****
121E 12C4 REM **** SCANNER PROGRAM # : ALL ****
121E 12C4 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
121E 12C4 REM **** SERVERS AS A WAIT AND REPLY ****
121E 12C4 REM **** ENTRY MODULE ****
121E 12C4 REM **** INPUT : SINGLE KEYBOARD ENTRY ****
121E 12C4 REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
121E 12C4 REM **** RESERVED LINE ****
121E 12C4 REM **** NUMBERS : 2001-2010 ****
121E 12C4 REM *****
121E 12C4 2001 REM REPLY FUNCTION
121F 12C4 2002 REPLY$=INKEY$ : IF REPLY$="" THEN 2002
1233 12C8 REPLY=ASC(REPLY$)
123D 12CA IF REPLY > 90 THEN REPLY$=CHR$(REPLY XOR 32) 'CONVERT TO CAPS
1258 12CA IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$="?"
1284 12CA RETURN
1287 12CA
1287 12CA 5000 REM $INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB
1288 12CA REM *****
1288 12CA REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1288 12CA REM **** SKIP COLE ****
1288 12CA REM **** SUBROUTINE NAME : RXXS5000.SUB ****
1288 12CA REM **** SCANNER PROGRAM # : ALL ****
1288 12CA REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1288 12CA REM **** PERFORMS A DATE EDIT ****
1288 12CA REM **** INPUT : DATE TO BE CHECKED MUST BE ****
1288 12CA REM **** IN THE VARIABLE NAMED ****
1288 12CA REM **** 'CK.5000$' ****
1288 12CA REM **** IN THE FORMAT "YYMMDD" ****
1288 12CA REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
1288 12CA REM **** VARIABLE. IF THIS VARIABLE ****
1288 12CA REM **** CONTAINS ANY NUMBER OTHER ****
1288 12CA REM **** THAN 0, AN ERROR WAS FOUND ****
1288 12CA REM **** IN THE DATE. ****
1288 12CA REM **** RESERVED LINE ****
1288 12CA REM **** NUMBERS : 5001-5009 ****
1288 12CA REM *****
1288 12CA RT.5000 = 0
128F 12CA CKYEAR = VAL(LEFT$(CK.5000$,2)) 'YEAR NUMERIC VALUE
12A2 12CC CKMONTH = VAL(MID$(CK.5000$,3,2)) 'MONTH NUMERIC VALUE
1288 12CE CKDAY = VAL(RIGHT$(CK.5000$,2)) 'DAY NUMERIC VALUE
12CB 12D0
12CB 12D0 IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
12E1 12D0 IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
12F7 12D0 IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
130D 12D0 IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
1323 12D0

```

```

Offset  Data    Source Line
1323  12D0  REM    LEAP YEAR CHECK
1323  12D0      MOLENGTH(2) = 28
132A  12D0      IF CKMONTH<> 2      THEN GOTO 5005  'MUST BE FEBRUARY
1339  12D0      IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005  'MUST BE A LEAP YEAR
134E  12D0      MOLENGTH(2) = 29
1355  12D0
1355  12D0  5005  IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
1374  12D0
1374  12D0  5009  RETURN
1377  12D0
1377  12D0  REM -----END OF SUBROUTINE 5000 -----
1377  12D0
1377  12D0  5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTP UCA VALIDATE SUB
1378  12D0  5600 REM INCLUDE: 'RACS5600.SUB' INCLUDE THE INP UCA VALIDATE SUB
1379  12D0
1379  12D0  6000 REM $INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE SUB
137A  12D0  REM *****
137A  12D0  REM *****  AMBULATORY CARE DATA BASE          13 APR 85      ****
137A  12D0  REM *****                                SKIP COLE          ****
137A  12D0  REM *****  SUBROUTINE NAME      :    RXXS6000.SUB          ****
137A  12D0  REM *****  SCANNER PROGRAM #    :    ALL                      ****
137A  12D0  REM *****  FUNCTION              :    THIS SUBROUTINE MODULE  ****
137A  12D0  REM *****                                PERFORMS INSTRING SEARCH ****
137A  12D0  REM *****                                ****
137A  12D0  REM *****  INPUT                  :    STRING TO BE SEARCHED MUST ****
137A  12D0  REM *****                                BE IN THE VARIABLE NAMED : ****
137A  12D0  REM *****                                'X$'                      ****
137A  12D0  REM *****                                ****
137A  12D0  REM *****  OUTPUT                  :    'TOT' = TOTAL NUMBER OF ****
137A  12D0  REM *****                                HITS IN THE DESTING ****
137A  12D0  REM *****                                'HOLD$( )' IS THE ARRAY ****
137A  12D0  REM *****                                CONTAINING THE NUMERIC ****
137A  12D0  REM *****                                VALUE OF THE HIT POSITIONS ****
137A  12D0  REM *****                                ****
137A  12D0  REM *****
137A  12D0  REM *****  RESERVED LINE                                ****
137A  12D0  REM *****  NUMBERS      :    6001-6009                      ****
137A  12D0  REM *****
137A  12D0  6001      PTR = INSTR(X$, "1")
1388  12D0      TOT = 0
138F  12D2      WHILE PTR > 0
139A  12D2          TOT=TOT+1
13A2  12D2          HOLD$(TOT) = RIGHT$(STR$(PTR),2)
13C4  12D2          PTR=PTR+1
13CC  12D2          PTR = INSTR(PTR,X$, "1")
13DE  12D2      WEND
13E2  12D2      RETURN
13E5  12D2
13E5  12D2  REM -----END OF SUBROUTINE RXXS6000.SUB-----
13E5  12D2
13E5  12D2  7000 REM $INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
13E6  12D2  REM *****

```

RACP850

GROUP FORM I & II DESTING/DECODE PROGRAM

PAGE 18

07-06-37

14:47:41

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

13E6 12D2 REM **** AMBULATORY CARE DATA BASE      13 APR 85      ****
13E6 12D2 REM ****                                SKIP COLE      ****
13E6 12D2 REM **** SUBROUTINE NAME      : RACS7000.SUB      ****
13E6 12D2 REM **** SCANNER PROGRAM #    : ALL                ****
13E6 12D2 REM **** FUNCTION              : THIS SUBROUTINE MODULE ****
13E6 12D2 REM ****                                PRINTS THE STANDARD SCREEN ****
13E6 12D2 REM ****                                HEADING.      ****
13E6 12D2 REM **** INPUT                  : COMMON VARIABLE USERS(2) ****
13E6 12D2 REM ****                                SYSTEM DATE    ****
13E6 12D2 REM ****                                ****
13E6 12D2 REM **** OUTPUT                  : SCREEN HEADING    ****
13E6 12D2 REM ****                                ****
13E6 12D2 REM **** RESERVED LINE          ****
13E6 12D2 REM **** NUMBERS                  : 7001-7010      ****
13E6 12D2 REM ****                                ****
13E6 12D2 7001 LOCATE 1,1
13F0 12D2 PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
13F8 12D2 LOCATE 1,65
1405 12D2 PRINT DATES;
140D 12D2 LOCATE 2,1
141A 12D2 PRINT "USER : ";USERS(1)
1427 12D2 RETURN
142A 12D2
142A 12D2 7100 REM $INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
142B 12D2 REM ****
142B 12D2 REM **** AMBULATORY CARE DATA BASE      13 APR 85      ****
142B 12D2 REM ****                                SKIP COLE      ****
142B 12D2 REM **** SUBROUTINE NAME      : RXXS7100.SUB      ****
142B 12D2 REM **** SCANNER PROGRAM #    : ALL                ****
142B 12D2 REM **** FUNCTION              : THIS SUBROUTINE MODULE ****
142B 12D2 REM ****                                PRINTS THE STANDARD HEADING ****
142B 12D2 REM ****                                ON THE PRINTER. ****
142B 12D2 REM **** INPUT                  : DATE,PAGE,PGMID$,PGMTITL$ ****
142B 12D2 REM ****                                ****
142B 12D2 REM **** OUTPUT                  : PRINTER HEADING, LN.COUNT ****
142B 12D2 REM ****                                ****
142B 12D2 REM **** RESERVED LINE          ****
142B 12D2 REM **** NUMBERS                  : 7101-7110      ****
142B 12D2 REM ****                                ****
142B 12D2 7101 IF PAGE > 0 THEN LPRINT CHR$(12);
1441 12D2 LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITL$;
144E 12D6 LPRINT TAB(70);DATES
1461 12D6 PAGE=PAGE+1
1469 12D6 LPRINT "PROGRAM ";PGMID$;TAB(70);"PAGE";
1486 12D6 LPRINT USING "####";PAGE
1492 12D6 LPRINT
149A 12D6 LN.COUNT=3
14A1 12D6 RETURN
14A4 12D6
14A4 12D6 8000 REM $INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
14A5 12D6 REM ****
14A5 12D6 REM **** AMBULATORY CARE DATA BASE      13 APR 85      ****

```

```

Offset  Data    Source Line
14A5 12D6 REM **** SKIP COLE ***
14A5 12D6 REM **** SUBROUTINE NAME : RXXS8000.SUB ***
14A5 12D6 REM **** SCANNER PROGRAM # : ALL ***
14A5 12D6 REM **** FUNCTION : THIS SUBROUTINE MODULE ***
14A5 12D6 REM **** IS A GROUPING THAT PERFORMS ***
14A5 12D6 REM **** VARIOUS DECODING FUNCTIONS ***
14A5 12D6 REM **** ON THE SCANNER DATA ***
14A5 12D6 REM ****
14A5 12D6 REM **** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
14A5 12D6 REM **** 8050 - CHECK FOR END OF JOB ***
14A5 12D6 REM **** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
14A5 12D6 REM **** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
14A5 12D6 REM **** (RETURNED IN TEXT$ STRING VARIABLE) ***
14A5 12D6 REM ****
14A5 12D6 REM **** INPUT : SHEET RECORD, RECORD LENGTH ***
14A5 12D6 REM ****
14A5 12D6 REM **** OUTPUT : 'TEXT$' TRING VARIABLE ***
14A5 12D6 REM ****
14A5 12D6 REM **** RESERVED LINE ***
14A5 12D6 REM **** NUMBERS : 8001-8500 ***
14A5 12D6 REM ****
14A5 12D6 'DECODE THE HEADER ONLY
14A5 12D6 8001 POINTER = 0
14AC 12D8 RECORDPTR = VARPTR(SHEETREC(0))
14B3 12DA FOR J8000 = 1 TO 21
14BA 12DA 8002 TEXT$= TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
14D8 12DA POINTER=POINTER+1
14E0 12DA NEXT J8000
14EF 12DC PROGRAM$= LEFT$(TEXT$,3)
14FE 12DC BATCH$= MID$(TEXT$,4,3)
1510 12E0 SERIAL$= MID$(TEXT$,7,4)
1522 12E4 RUNID$= MID$(TEXT$,11,1)
1534 12E8 FORM$= MID$(TEXT$,12,2)
1546 12EC POCKET$= MID$(TEXT$,14,1)
1558 12F0 SCANERR1$=MID$(TEXT$,16,2)
156A 12F4 SCANERR2$=MID$(TEXT$,18,2)
157C 12F8 SCANERR3$=MID$(TEXT$,20,2)
158E 12FC GOTO 8500
1592 12FC
1592 12FC 8050 REM CHECK FOR END OF JOB/END OF BATCH
1593 12FC IF PROGRAM$ = PGMID$ THEN GOTO 8500
15A5 12FC LPRINT STRING$(80,"")
15B3 12FC LPRINT
15B8 12FC LPRINT "RECORDS PROCESSED ... ";SERIAL$
15C8 12FC LPRINT "STARTED AT ..... ";BTIM$
15D5 12FC LPRINT "ENDED AT ..... ";TIM$
15E2 12FC LPRINT CHR$(12)
15E0 12FC GOTO 30000
15F1 12FC
15F1 12FC 8070 REM CHECK FOR SCANNER ERRORS
15F2 12FC IF POCKET$ = " " GOTO 8500
1604 12FC LPRINT LITHOS;
160C 12FC LPRINT " ... SCANNER ERRORS : ";

```

Offset	Data	Source Line
1614	12FC	LPRINT SCANERR1\$;" / ";
1621	12FC	LPRINT SCANERR2\$;" / ";
162E	12FC	LPRINT SCANERR3\$
1636	12FC	LN=LN+1
163E	12FE	GOTO 999
1642	12FE	
1642	12FE	8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
1643	12FE	LOCATE 5,1:PRINT "PROGRAM ";PROGRAM\$;
1650	12FE	
1650	12FE	PRINT " BATCH ";BATCH\$;
166A	12FE	PRINT " RUN ";RUNID\$;
1677	12FE	PRINT " FORM ";FORM\$;
1684	12FE	PRINT " POCKET ";POCKET\$
1691	12FE	GOTO 8500
1695	12FE	
1695	12FE	8200 REM DECODE THE RESPONSE POSITIONS
1696	12FE	POINTER = 21
169D	12FE	RECORDPTR = VARPTR(SHEETREC(0))
16A4	12FE	FOR J8000 = 22 TO RECORDLENGTH
16B1	1300	8202 TEXT\$ = TEXT\$+CHR\$(PEEK(RECORDPTR + POINTER))
16CF	1300	POINTER=POINTER+1
16D7	1300	NEXT J8000
16E8	1300	
16E8	1300	8500 RETURN
16EB	1300	
16EB	1300	REM ----- END OF RXXS8000.SUB -----
16EB	1300	
16EB	1300	9000 REM \$INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
16EC	1300	REM *****
16EC	1300	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
16EC	1300	REM **** SKIP COLE ****
16EC	1300	REM **** PROGRAM NAME : RACS9000.SUB ****
16EC	1300	REM **** SCANNER PROGRAM # : ALL ****
16EC	1300	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
16EC	1300	REM **** CONTROLS THE SCANNER I/O ****
16EC	1300	REM ****
16EC	1300	REM **** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS ****
16EC	1300	REM **** COMMUNICATIONS MANUAL AND THE ****
16EC	1300	REM **** PRE-RELEASED SOFTWARE GUIDE ****
16EC	1300	REM ****
16EC	1300	REM *****
16EC	1300	REM **** RESERVED LINE ****
16EC	1300	REM **** NUMBERS : 9001-9100 ****
16EC	1300	REM *****
16EC	1300	
16EC	1300	REM *****
16EC	1300	REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
16EC	1300	REM **** ARGUMENTS: PRESET ... SEE BELOW ****
16EC	1300	REM *****
16EC	1300	9001 REM
16ED	1300	PROTOCOL(0) = 9600 'BAUD RATE
16F4	1300	PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
16FB	1300	PROTOCOL(2) = 8 'DATA BITS

Offset	Data	Source Line
1702	1300	PROTOCOL(3) = 1 'STOP BITS
1709	1300	PROTOCOL(4) = 2 'RS-232 PORT
1710	1300	PROTOCOL(5) = 0 'WRITE TIME-OUT
1717	1300	PROTOCOL(6) = 0 'READ TIME-OUT
171E	1300	
171E	1300	ERRSTAT\$ = SPACES(60)
172A	1300	ARGPTR = VARPTR(PROTOCOL(0))
1731	1302	CALL SETUP (ARGPTR,ERRSTAT\$)
1742	1302	ERRMSG\$=""
1748	1302	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="SETUP ERROR "+ERRSTAT\$
1767	1302	GOTO 9100
1768	1302	
1768	1302	REM *****
1768	1302	REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
1768	1302	REM **** ARGUMENTS: CNTRLOPT ****
1768	1302	REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
1768	1302	REM **** CNTRLOPT = 2 = STOP SCANNER (S0) ****
1768	1302	REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
1768	1302	REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
1768	1302	REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
1768	1302	REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
1768	1302	REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
1768	1302	REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
1768	1302	REM *****
1768	1302	9010 REM
176C	1302	ERRSTAT\$ = SPACES(60)
1778	1302	CALL CNTRLOP (CNTRLOPT,ERRSTAT\$)
1789	1302	ERRMSG\$=""
1792	1302	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
17AE	1302	GOTO 9100
17B2	1302	
17B2	1302	REM *****
17B2	1302	REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
17B2	1302	REM ****
17B2	1302	REM **** ARGUMENTS: READTYPE ****
17B2	1302	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
17B2	1302	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
17B2	1302	REM ****
17B2	1302	REM **** ARGUMENTS: RECORDLENGTH ****
17B2	1302	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
17B2	1302	REM **** TRANSMITTED ****
17B2	1302	REM *****
17B2	1302	9020 REM
17B3	1302	ERRSTAT\$ = SPACES(60)
17BF	1302	RECORDPTR = VARPTR(SHEETREC(0))
17C6	1302	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
17DF	1302	ERRMSG\$=""
17E8	1302	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
1809	1302	GOTO 9100
180D	1302	
180D	1302	REM *****
180D	1302	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
180D	1302	REM ****
180D	1302	REM **** ARGUMENTS: PRINTPOS ****

RACP850

GROUP FORM I & II DESTING/DECODE PROGRAM

PAGE 22

07-06-87

14:47:41

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
1800	1302	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
1800	1302	REM **** VALUES = 0 THRU 90 ****
1800	1302	REM ****
1800	1302	REM **** ARGUMENTS: PSTRINGS ****
1800	1302	REM **** TEXT TO BE PRINTED ON THE FORM ****
1800	1302	REM ****
1800	1302	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
1800	1302	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
1800	1302	REM ****
1800	1302	9030 REM
180E	1302	ERRSTAT\$ = SPACES(60)
181A	1302	RECORDPTR = VARPTR(SHEETREC(0))
1821	1302	CALL TPRINT(PRINTPOS,PSTRINGS,ERRSTAT\$)
1836	1308	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
1852	1308	GOTO 9100
1856	1308	
1856	1308	9100 RETURN
1859	1308	REM -----END OF SUBROUTINE RACS9000.SUB -----
1859	1308	
1859	1308	REM END OF SUBROUTINES =====
1859	1308	
1859	1308	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
185A	1308	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1862	1308	LPRINT "ERASING FILE ";DATFIL\$
186F	1308	BEEP
1873	1308	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
187F	1308	CLOSE
1883	1308	OPEN DATFIL\$ FOR OUTPUT AS #1
1895	1308	PRINT #1,STRING\$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
18A7	1308	CLOSE
18AB	1308	
18AB	1308	30000 REM
18AC	1308	CLOSE
18B0	1308	CHAIN "RACP10"
18B7	1308	
18BA	1308	

22151 Bytes Available

14696 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

PAGE 1
04-24-87
12:51:22

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

001A 0002 REM \$LINESIZE: 132
001A 0002 REM \$PAGESIZE: 66
001A 0002 REM \$TITLE: 'RACP860 '
001A 0002 REM \$SUBTITLE: 'IMMUNIZATION SHORT FORM'
001A 0002 REM \$PAGE

IMMUNIZATION SHORT FORM

04-24-87

12:51:22

IBM Personal Computer BASIC Compiler V1.00

```

Offset  Data    Source Line
-----
001A 0002 REM +-----+
001A 0002 REM | NAME: RACPB60          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | DATE: 15 APR 87      IMMUNIZATION                     |
001A 0002 REM | D R BOLLING          SHORT FORM                         |
001A 0002 REM |
001A 0002 REM | INCLUDES PREFIX TO LITHO FOR EACH PATIENT ON FORM      |
001A 0002 REM +-----+
001A 0002 REM          IMMUN SHORT FORM INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the SHORT form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RXXP05, otherwise,
001A 0002 REM the program chains to program RXXP10 on exit.
001A 0002 REM
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD          DIMENSION DEFINITIONS      *
001A 0002 '* Date: 28 Feb 84          Written by: Floyd Cole      *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' effect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM....
001A 0002
001A 0002 DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 DIM ED.MSG$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002 DIM CLINIC1.PFX$(5) '(PREFIX -B D F G S- FOR CLINIC #1)
001A 0002 DIM PROCES$(39) '(PROCEDURE TABLE FOR SHORT FORM)
001A 0002 DIM PTID$(6) '(PATIENT SSN+PMP)

```

```

Offset  Data    Source Line
001A 0002      DIM TOT.PROC(6)      '(PROCEDURE TOTAL PER PATIENT)
001A 0002      DIM GROUP$(6,39)    '(PATIENT/PROCEDURE GROUP)
001A 0002      DIM SPECDS$(6)      '(SPECIFIC PRE CODE GROUP)
001A 0002      DIM HOLDS$(39)      '(HOLD AREA FOR SUBROUTINE 6000)
001A 0002      REM $INCLUDE: 'RACCMN.MOD'      REM INCLUDE THE COMMON AREA DEFINITION
001A 0002      *****
001A 0002      '* NAME: RACCMN.MOD              COMMON AREA DEFINITION      *
001A 0002      '* Date: 28 Feb 84              Written by: Floyd Cole      *
001A 0002      *****
001A 0002      ' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A 0002      ' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A 0002      '
001A 0002      ' This program segment may be modified, but all files containing
001A 0002      ' an include for this segment must be re*compiled in order to
001A 0002      ' affect the changes made here.
001A 0002      '
001A 0002      ' *****START OF COMMON DEFINITIONS*****
001A 0002      COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL$ 'BASIC SCREEN COLORS
001A 0002      COMMON HEADERS$ '21 CHARACTER SCANNER HEADER INFO
001A 0002      COMMON TEXT$ '11 AINING CHARACTERS FROM SCANNER
001A 0002      COMMON PGMID$ 'PROGRAM OR FORM ID
001A 0002      COMMON MOLENGTH() 'DAYS IN THE MONTH
001A 0002      COMMON USERS$()
001A 0002      ' *****END OF COMMON DEFINITION*****
001A 0002
001A 0002      REM $INCLUDE: 'RACDEF.MOD'      REM INCLUDE THE DEFAULT DEFINITIONS
001A 0002      *****
001A 0002      '* NAME: RACP01.DEF              DEFAULT DEFINITIONS      *
001A 0002      '* Date: 28 Feb 84              Written by: Floyd Cole      *
001A 0002      *****
001A 0002      ' Variables used in common that have a default value on start*up
001A 0002      ' will be held in this file. It is an included file so it cannot
001A 0002      ' be run in a stand*alone mode. In normal operation, this file
001A 0002      ' should be 'included' in the main program only (RACP10.BAS).
001A 0002      '
001A 0002      ' This program segment may be modified, but all files containing
001A 0002      ' an include for this segment must be re*compiled in order to
001A 0002      ' affect the changes made here.
001A 0002      '
001A 0002      ' *****START OF DEFAULT DEFINITION*****
001A 0002      FORE = 15      'FOREGROUND COLOR = INTENSE WHITE
0047 144A      BACK = 1      'Background Color = Light Blue
004E 144A      BORD = 4      'BORDER = RED
0055 144C      HIDE = 4      'ALTERNATE COLOR = RED
005C 144C      EFORE= 14      'ERROR FOREGROUND DISPLAY
0063 144C      EBACK= 0      'ERROR BACKGROUND DISPLAY
006A 144C      BELL$ = CHR$(7) 'Sound the bell
0076 144C
0076 144C      MOLENGTH(1) = 31      'JAN
007D 144C      MOLENGTH(2) = 28      'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB

```

Offset	Data	Source Line
0084	144C	MOLENGTH(3) = 31 'MAR
008B	144C	MOLENGTH(4) = 30 'APR
0092	144C	MOLENGTH(5) = 31 'MAY
0099	144C	MOLENGTH(6) = 30 'JUN
00A0	144C	MOLENGTH(7) = 31 'JUL
00A7	144C	MOLENGTH(8) = 31 'AUG
00AE	144C	MOLENGTH(9) = 30 'SEP
00B5	144C	MOLENGTH(10) = 31 'OCT
00BC	144C	MOLENGTH(11) = 30 'NOV
00C3	144C	MOLENGTH(12) = 31 'DEC
00CA	144C	
00CA	144C	DATEERR\$(0) = " "
00D3	144C	DATEERR\$(1) = "INVALID MONTH"
00DC	144C	DATEERR\$(2) = "INVALID DAY "
00E5	144C	DATEERR\$(3) = "DAY TOO LARGE FOR MONTH CODED"
00EE	144C	
00EE	144C	MAXLENGTH = 80 'MAXIMUM LENGTH OF OUTPUT RECORD
00F5	144E	PADS = " " 'PAD CHARACTER FOR SHORT RECORDS
00FE	1452	
00FE	1452	' *****END OF DEFAULT DEFINITION*****
00FE	1452	
00FE	1452	KEY OFF
0104	1452	
0104	1452	REM *****
0104	1452	REM THE FOLLOWING VARIABLES ARE UNIQUE TO EACH PROGRAM AND MUST
0104	1452	REM BE CHANGED.
0104	1452	REM *****
0104	1452	PGMTITL\$ = "IMMUNIZATION SHORT FORM"
010D	1456	
010D	1456	PGMID\$ = "860" 'VALUE RECEIVED FROM THE SCANNER
0116	1456	'IN HEADER VARIABLE 'PROGRAM\$'
0116	1456	
0116	1456	DATFIL\$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
011F	145A	
011F	145A	REM LENGTH OF STRING RECEIVED FROM THE OMR....
011F	145A	HEADER = 21
0126	145C	RESPONSE= 376
012D	145E	RECORDLENGTH = HEADER + RESPONSE
0138	1460	
0138	1460	N.PROC = 39 'NUMBER OF PROCEDURES FOR THIS FORM
013F	1462	
013F	1462	REM *****
013F	1462	
013F	1462	BTIMES=TIMES 'SCAN START TIME
0148	1466	
0148	1466	
0148	1466	REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0148	1466	REM CLINIC #1
0148	1466	CLINIC1.PFX\$(0)=" "
0151	1466	CLINIC1.PFX\$(1)="B"
015A	1466	CLINIC1.PFX\$(2)="D"
0163	1466	CLINIC1.PFX\$(3)="F"
016C	1466	CLINIC1.PFX\$(4)="G"

Offset	Data	Source Line
0175	1466	CLINIC1.PFX\$(5)="S"
017E	1466	
017E	1466	REM *** ENCOUNTER FORM PROCEDURE TABLE ***
017E	1466	PROCD\$(0)=" " : PROCD\$(14) = "90724" : PROCD\$(28)= "86580"
0199	1466	PROCD\$(1)="90723" : PROCD\$(15) = "90705" : PROCD\$(29)= "86581"
01B4	1466	PROCD\$(2)="90725" : PROCD\$(16) = "90733" : PROCD\$(30)= "86582"
01CF	1466	PROCD\$(3)="90701" : PROCD\$(17) = "90707" : PROCD\$(31)= "90703"
01EA	1466	PROCD\$(4)="90702" : PROCD\$(18) = "90704" : PROCD\$(32)= "90714"
0205	1466	PROCD\$(5)="90718" : PROCD\$(19) = "90727" : PROCD\$(33)= "90717"
0220	1466	PROCD\$(6)="90731" : PROCD\$(20) = "90732" : PROCD\$(34)= "90700"
0238	1466	PROCD\$(7)="90711" : PROCD\$(21) = "90712" : PROCD\$(35)= "90698"
0256	1466	PROCD\$(8)="90742" : PROCD\$(22) = "90713" : PROCD\$(36)= "95120"
0271	1466	PROCD\$(9)="90741" : PROCD\$(23) = "90726" : PROCD\$(37)= "95122"
028C	1466	PROCD\$(10)="90743" : PROCD\$(24) = "90708" : PROCD\$(38)= "95123"
02A7	1466	PROCD\$(11)="90744" : PROCD\$(25) = "90709" : PROCD\$(39)= "95124"
02C2	1466	PROCD\$(12)="90746" : PROCD\$(26) = "90706" :
02D4	1466	PROCD\$(13)="90747" : PROCD\$(27) = "90710" :
02E6	1466	
02E6	1466	REM INCLUDE: 'UCABAMC.OPT' INCLUDE OUTPATIENT UCA TABLE
02E6	1466	
02E6	1466	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
02EB	1466	CLS
02EF	1466	GOSUB 7000 'PRINT SCREEN HEADING
02F4	1466	
02F4	1466	REM *****
02F4	1466	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
02F4	1466	REM *****
02F4	1466	REM
02F4	1466	OPEN DATFIL\$ FOR APPEND AS #1
0306	1466	
0306	1466	REM *****
0306	1466	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
0306	1466	REM *****
0306	1466	LPRINT CHR\$(15);
0311	1466	WIDTH "LPT1:",160
0318	1466	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
0327	1468	COLOR 14
032E	1468	LOCATE 11,26 : PRINT "IMMUNIZATION SHORT FORM"
0343	1468	COLOR FORE,BACK,BORD
0359	1468	
0359	1468	REM *****
0359	1468	REM **** COMMUNICATIONS SETUP ****
0359	1468	REM *****
0359	1468	REM PROTOCOL
0359	1468	GOSUB 9001
035E	1468	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0378	146C	
0378	146C	REM START SCANNER (S1)
0378	146C	CNTRLOPT =1 :GOSUB 9010
0384	146E	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
039E	146E	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
0393	146E	READYTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
03BA	1470	

Offset	Data	Source Line
038A	1470	REM *****
038A	1470	REM **** SET SCAN SHEET CALL ****
038A	1470	REM *****
038A	1470	REM
038A	1470	
038A	1470	10 REM - RETURN POINT TO READ NEXT SHEET
038B	1470	
038B	1470	AS=INKEY\$
03C4	1474	IF AS=CHR\$(27) THEN GOTO 25000
03DA	1474	
03DA	1474	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
03DF	1474	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
03FB	1478	
03FB	1478	TEXT\$="" 'CLEAR THE INPUT AREA
0404	1478	GOSUB 8000 'DECODE HEADER
0409	1478	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
040E	1478	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
0413	1478	LITHO\$ = MID\$(TEXT\$,22,8)
0425	147C	GOSUB 8070 'CHECK FOR SCANNER ERRORS
042A	147C	GOSUB 8100 'PRINT THE DATA ON THE SCREEN
042F	147C	
042F	147C	REM \$INCLUDE: 'RACH860.MO1' INCLUDE THE SHORT FORM REFORMAT/EDIT MOD
042F	147C	REM *****
042F	147C	REM **** AMBULATORY CARE INFORMATION SYSTEM 15 APR 87 ****
042F	147C	REM **** D R BOLLING ****
042F	147C	REM **** MODULE NAME : RACH860.MO1 ****
042F	147C	REM **** SCANNER PROGRAM # : 860-IMMUNIZATION (FBIA) FORM ****
042F	147C	REM ****
042F	147C	REM **** PURPOSE : REFORMAT/EDIT THE ENCOUNTER ****
042F	147C	REM **** SHORT FORM OMR RECORD. ****
042F	147C	REM *****
042F	147C	REM **** RESERVED LINE NUMBERS 100-199 ****
042F	147C	REM *****
042F	147C	
042F	147C	N.ERR =0 'COUNTS THE NUMBER OF ERRORS
0436	147E	
0436	147E	REM *** LITHO CODE DONE IN BAS PROGRAM ***
0436	147E	
0436	147E	REM *** CLINIC ID ***
0436	147E	100 CL1.COD\$="FBIA" 'DEFAULT CLINIC CODE
043F	1482	
043F	1482	REM ... CLINIC CODE WHEN HAND CODED ...
043F	1482	IF MID\$(TEXT\$,30,4)=" " THEN 104
0457	1482	CK.X=VAL(MID\$(TEXT\$,30,1))
046D	1484	CK.COD\$=MID\$(TEXT\$,31,3)
047F	1488	REM GOSUB 5500 'VALIDATE OUTP UCA CODE
047F	1488	REM IF RT.5500 = 0 THEN GOTO 101
047F	1488	REM N.ERR = N.ERR + 1
047F	1488	REM ED.MSG\$(N.ERR)="*1-2 INVALID CLINIC CODE"
047F	1488	
047F	1488	CL1.COD\$=CLINIC1.PFX\$(CK.X) + CK.COD\$
0497	1488	REM *** VISIT DATE ***
0497	1488	104 CMS=MID\$(DATE\$,1,2) 'CURRENT MONTH

Offset Data Source Line

```

04A9 148C      YRS=MID$(DATE$,9,2)      'CURRENT YEAR
04B8 1490      X$=MID$(TEXT$,35,4)      'MONTH AND DAY FROM FORM
04CD 1494      IF LEFT$(X$,2)<=CH$ THEN 105  'OK, USE THIS YEAR
04E2 1494      YRS=RIGHT$(STR$(VAL(YRS)-1),2) 'USE LAST YEAR
0502 1494      105  VDATES = YRS + X$
0510 1498      'EDIT VISIT DATE
0510 1498      CK.5000$=VDATES
0519 149C      GOSUB 5000      'DATE CHECK
051E 149C      CK.5010$=VDATES
0527 14A0      GOSUB 5010      'NUMERIC STRING CHECK
052C 14A0      IF RT.5000 = 0 AND RT.5010 = 0 THEN GOTO 106
0550 14A4      N.ERR=N.ERR+1
0558 14A4      ED.MSG$(N.ERR)="TODAYS DATE " + DATEERR$(RT.5000)
057E 14A4
057E 14A4      REM *** PROVIDER ID (PREFIX + NUM) ***
057E 14A4      106  PROV1.PFX$ = MID$(TEXT$,39,1)
0590 14A8      PROV1.NUM$ = MID$(TEXT$,40,4)
05A2 14AC
05A2 14AC      SSN.OFFSET = 44
05A9 14AE      FMP.OFFSET = 53
0580 1480      PRC.OFFSET = 55
0587 1482      SPE.OFFSET = 94
058E 1484      TOT.OFFSET = 59
05C5 1486      POINTER = 0
05CC 1488
05CC 1488      REM LOOP THROUGH THE FOLLOWING CODE 6 TIMES (1 FOR EACH GROUP)
05CC 1488      FOR I411 = 1 TO 6
05D3 1488      REM PATIENT ID (SSN+FMP)
05D3 1488      110  SSN$=MID$(TEXT$,SSN.OFFSET+POINTER,9)
05EA 148C      FMP$=MID$(TEXT$,FMP.OFFSET+POINTER,2)
0601 14C0
0601 14C0      PTID$(I411) = SSN$+FMP$
061A 14C2      IF I411=1 THEN GOTO 120
0629 14C2      IF PTID$(I411)=STRING$(11," ") THEN GOTO 130
064F 14C2
064F 14C2      REM *** PROCEDURE CODES ***
064F 14C2      120  X$ = MID$(TEXT$,PRC.OFFSET+POINTER,N.PROC)
0667 14C2      GOSUB 6000      'DECODE THE X$ STRING
066C 14C2      IF TOT = 0 THEN GOTO 122
0678 14C4      TOT.PROC(I411) = TOT
068A 14C4      FOR I411.A= 1 TO TOT
0697 14C6      PTR=VAL(HOLD$(I411.A))
06AD 14CA      GROUP$(I411,I411.A)=PROCED$(PTR)
06CF 14CA      NEXT I411.A
06E0 14CA      GOTO 124
06E4 14CA
06E4 14CA      122  N.ERR=N.ERR+1
06EC 14CA      ED.MSG$(N.ERR)="NO PROCEDURE CODE "
0700 14CA
0700 14CA      REM *** SPECIFIC PREASSIGNED CODES ***
0700 14CA      124  INP.STO$=MID$(TEXT$,SPE.OFFSET+POINTER,9)
0717 14CE
0717 14CE      GOSUB 5700      'CONVERT ARRAY STRING
071C 14CE      SPECDS$(I411)=BUF.STO$      'UP TO 9 TWO DIGIT CODES

```

```

Offset  Data    Source Line
-----  -
0730  1402
0730  1402      130  POINTER = POINTER + TOT.OFFSET
073B  1402          NEXT 1411
0740  1402
0740  1402      199  REM
074E  1402
074E  1402      REM -----END OF MODULE   RACH860.M01-----
074E  1402
074E  1402          IF N.ERR = 0 THEN GOTO 997
075D  1402          LPRINT "LITHO # ";LITHOS;" ... ERRORS"
076F  1402          FOR 1997 = 1 TO N.ERR
077C  1404              LPRINT USING "### ";1997;
0788  1406              LPRINT "=> ";ED.MSG$(1997)
079E  1406              NEXT 1997
07AF  1406          LN.COUNT = LN.COUNT + N.ERR + 1
078B  1408          CNTRLOPT = 6
07C2  1408          GOSUB 9010          'REJECT THE FORM
07C7  1408          GOTO 998          'BYPASS THE DISK WRITER....
07CB  1408
07CB  1408      997  REM $INCLUDE: 'RACH860.M02'      REM INCLUDE THE BASE ENCOUNTER FORM DISK WRITER
07CC  1408      REM *****
07CC  1408      REM *****  AMBULATORY CARE INFORMATION SYSTEM      15 APR 87      *****
07CC  1408      REM *****                                D R BOLLING *****
07CC  1408      REM *****  MODULE NAME      :      RACH860.MOD      *****
07CC  1408      REM *****  SCANNER PROGRAM #  :      860-IMMUNIZATION SHORT FORM *****
07CC  1408      REM *****
07CC  1408      REM *****  PURPOSE      :      CREATE AND WRITE THE DISK *****
07CC  1408      REM *****                                RECORD FOR INPUT TO FOCUS *****
07CC  1408      REM *****
07CC  1408      REM *****  PROGRAM ADDS PREFIX TO LITHO FOR EACH PATIENT *****
07CC  1408      REM *****
07CC  1408      REM *****  RESERVED LINE NUMBERS 200-299 *****
07CC  1408      REM *****
07CC  1408
07CC  1408      REM  BUILD THE OUTPUT RECORD
07CC  1408
07CC  1408          GOSUB 270          'BUILD THE RECORD KEY
07D1  1408
07D1  1408      REM *****
07D1  1408      REM *****  RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE *****
07D1  1408      REM *****
07D1  1408      REM  RECOUNT$ ="8602"+RECKEY$ 'TRANSACTION IDENTIFIER
07D1  1408          FRMNS$ = "86"          'FORM NUMBER
07DA  140C
07DA  140C          FOR 1412 = 1 TO 6
07E1  140C              IF PTID$(1412)=STRING$(11," ") THEN GOTO 250
0807  140E              FOR 1412.A = 1 TO TOT.PROC(1412)
081B  14E0                  RECOUNT$ = "8602" + RECKEY$ + RIGHT$(STR$(1412),1) + RIGHT$(LITHOS,7)
084B  14E8                  MID$(RECOUNT$,20,11)=PTID$(1412)
0863  14E8                  RECOUNT$ =RECOUNT$ + FRMNS$ + "1" + GROUP$(1412,1412.A)
088F  14EA                  GOSUB 280
0894  14EA                  PRINT #1,RECOUNT$
089F  14EA                  NEXT 1412.A

```

```

Offset Data Source Line
0883 14EA 250 NEXT 1412
08C5 14EA
08C5 14EA REM *****
08C5 14EA REM **** RECORD TYPE "3" - RECKEY PLUS SPEC PRE CLINIC CODES ****
08C5 14EA REM *****
08C5 14EA FOR 1860 = 1 TO 6
08CC 14EA IF LEN(SPECDS(1860))=0 THEN 260
08E2 14EC RPOINT = 1
08E9 14EE
08E9 14EE 252 RECOD3$=RIGHT$(MID$(SPECDS(1860),RPOINT,2),1)
0908 14F2 IF RECOD3$="" THEN 254
0919 14F2
0919 14F2 RECOUNT$="8603" + RECKEY$ + RIGHT$(STR$(1860),1) + RIGHT$(LITHOS,7)
0949 14F2 MID$(RECOUNT$,20,11)=PTIDS(1860)
0961 14F2 RECOUNT$=RECOUNT$ + FRMNS + RECOD3$
0974 14F2 GOSUB 280
0979 14F2 PRINT #1,RECOUNT$
0984 14F2
0984 14F2 254 RPOINT = RPOINT + 2
0980 14F2 IF RPOINT < LEN(SPECDS(1860)) THEN 252
09A5 14F2
09A5 14F2 260 NEXT 1860
0987 14F2
0987 14F2 REM END OF TYPE 3 RECORDS
0987 14F2
0987 14F2 GOTO 299
0988 14F2
0988 14F2 REM *****
0988 14F2 REM **** SUBROUTINE 270 - BUILD THE RECORD KEY ****
0988 14F2 REM *****
0988 14F2 270 RECKEY$=""
09C4 14F2
09C4 14F2 REM *** CLINIC ID (PREFIX + CODE) ***
09C4 14F2 RECKEY$= CL1.COD$
09CD 14F2
09CD 14F2 REM *** VISIT DATE ***
09CD 14F2 RECKEY$=RECKEY$+ VDATES
09D9 14F2
09D9 14F2 REM *** PROVIDER ID (PREFIX + NUM) ***
09D9 14F2 RECKEY$ = RECKEY$ + PROV1.PFX$ + PROV1.NUM$
09EC 14F2
09EC 14F2 REM *** PATIENT ID (SSN+FMP) ***
09EC 14F2 RECKEY$ = RECKEY$ + STRINGS(11," ")
09FE 14F2
09FE 14F2 REM *** LITHO CODE ***
09FE 14F2 REM DO LITHO IN LOOP ABOVE
09FE 14F2 REM RECKEY$ = RECKEY$ + LITHOS
09FE 14F2
09FE 14F2 RETURN
0A01 14F2
0A01 14F2 REM *****
0A01 14F2 REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
0A01 14F2 REM *****
0A01 14F2 280 PAD=MAXLENGTH - LEN(RECOUNT$) '0

```

12:51:22

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
0A0F	14F4	RECOUTS = RECOUTS + STRING\$(PAD,PADS) 'PAD THE RECORD WITH FILL CHAR
0A22	14F4	RETURN
0A25	14F4	
0A25	14F4	
0A25	14F4	299 REM
0A26	14F4	
0A26	14F4	REM -----END OF MODULE RXXM860.M02-----
0A26	14F4	
0A26	14F4	998 REM CONTINUE
0A27	14F4	
0A27	14F4	999 READTYPE = 2
0A2E	14F4	IF LN.COUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
0A3E	14F4	GOTO 10
0A42	14F4	
0A42	14F4	REM END OF SCAN/DECODE/WRITE LOOP =====
0A42	14F4	
0A42	14F4	1000 REM \$INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
0A43	14F4	REM =====
0A43	14F4	REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
0A43	14F4	REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
0A43	14F4	REM =====
0A43	14F4	REM PATIENT OMR INPUT PROGRAM *
0A43	14F4	REM *
0A43	14F4	REM This program verifies user is logged on properly. If there is no *
0A43	14F4	REM valid user logged on at the time of execution, this subroutine will*
0A43	14F4	REM chain to the logon program RACPO5, otherwise a return is issued. *
0A43	14F4	REM =====
0A43	14F4	REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
0A43	14F4	REM =====
0A43	14F4	1001 OPEN "I",1,"RACLOG.DAT"
0A55	14F4	IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
0A63	14F4	INPUT #1,USERS(1),DT\$,TMS,PIDS
0A84	1500	IF USERS(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
0A92	1500	IF USERS(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
0AA0	1500	CLOSE 1
0AA7	1500	SCREEN 0,1,0,0
0AB0	1500	COLOR FORE,BACK,BORD
0AD3	1500	CLS
0AD7	1500	RETURN
0ADA	1500	
0ADA	1500	1002 CLOSE
0ADE	1500	CHAIN "RACPO5"
0AE5	1500	'=====END OF LOGON VERIFY SUBROUTINE 1000=====
0AE5	1500	
0AE5	1500	2000 REM \$INCLUDE: 'RACS2000.SUB' INCLUDE THE REPLY/DELAY SUB
0AE6	1500	REM =====
0AE6	1500	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0AE6	1500	REM **** SKIP COLE ****
0AE6	1500	REM **** SUBROUTINE NAME : RACS2000.SUB ****
0AE6	1500	REM **** SCANNER PROGRAM # : ALL ****
0AE6	1500	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0AE6	1500	REM **** SERVERS AS A WAIT AND REPLY ****
0AE6	1500	REM **** ENTRY MODULE ****
0AE6	1500	REM **** INPUT : SINGLE KEYBOARD ENTRY ****

```

Offset  Data  Source Line
-----
0AE6 1500 REM ****
0AE6 1500 REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
0AE6 1500 REM ****
0AE6 1500 REM **** RESERVED LINE ****
0AE6 1500 REM **** NUMBERS : 2001-2010 ****
0AE6 1500 REM ****
0AE6 1500 2001 REM REPLY FUNCTION
0AE7 1500 2002 REPLY$=INKEY$ : IF REPLY$="" THEN 2002
0AFB 1504 REPLY$=ASC(REPLY$)
0B05 1506 IF REPLY > 90 THEN REPLY$=CHR$(REPLY XOR 32) 'CONVERT TO CAPS
0B20 1506 IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$="?"
0B4C 1506 RETURN
0B4F 1506
0B4F 1506 5000 REM $INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB
0B50 1506 REM *****
0B50 1506 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0B50 1506 REM **** SKIP COLE ****
0B50 1506 REM **** SUBROUTINE NAME : RXXS5000.SUB ****
0B50 1506 REM **** SCANNER PROGRAM # : ALL ****
0B50 1506 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0B50 1506 REM **** PERFORMS A DATE EDIT ****
0B50 1506 REM ****
0B50 1506 REM **** INPUT : DATE TO BE CHECKED MUST BE ****
0B50 1506 REM **** IN THE VARIABLE NAMED ****
0B50 1506 REM **** 'CK.5000$' ****
0B50 1506 REM **** IN THE FORMAT "YYMMDD" ****
0B50 1506 REM ****
0B50 1506 REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
0B50 1506 REM **** VARIABLE. IF THIS VARIABLE ****
0B50 1506 REM **** CONTAINS ANY NUMBER OTHER ****
0B50 1506 REM **** THAN 0, AN ERROR WAS FOUND ****
0B50 1506 REM **** IN THE DATE. ****
0B50 1506 REM ****
0B50 1506 REM **** RESERVED LINE ****
0B50 1506 REM **** NUMBERS : 5001-5009 ****
0B50 1506 REM *****
0B50 1506 RT.5000 = 0
0B57 1506 CKYEAR = VAL(LEFT$(CK.5000$,2)) 'YEAR NUMERIC VALUE
0B6A 1508 CKMONTH = VAL(MID$(CK.5000$,3,2)) 'MONTH NUMERIC VALUE
0B80 150A CKDAY = VAL(RIGHT$(CK.5000$,2)) 'DAY NUMERIC VALUE
0.93 150C
0B93 150C IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
0BA9 150C IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
0BBF 150C IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
0BD5 150C IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
0BEB 150C
0BEB 150C REM LEAP YEAR CHECK
0BEB 150C MOLENGTH(2) = 28
0BF2 150C IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY
0C01 150C IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR
0C16 150C MOLENGTH(2) = 29
0C1D 150C
0C1D 150C 5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
0C3C 150C

```

```

Offset Data Source Line
0C3C 150C 5009 RETURN
0C3F 150C
0C3F 150C REM -----END OF SUBROUTINE 5000 -----
0C3F 150C
0C3F 150C 5010 REM $INCLUDE: 'RACS5010.SUB' INCLUDE THE NUMERIC STRING EDITOR
0C40 150C REM *****
0C40 150C REM **** AMBULATORY CARE DATA BASE 1 MAY 85 ****
0C40 150C REM **** SKIP COLE ****
0C40 150C REM **** SUBROUTINE NAME : RXXS5010.SUB ****
0C40 150C REM **** SCANNER PROGRAM # : ALL ****
0C40 150C REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0C40 150C REM **** PERFORMS A NUMERIC STRING ****
0C40 150C REM **** EDIT. ****
0C40 150C REM ****
0C40 150C REM **** INPUT : STRING TO BE EDITED IS IN ****
0C40 150C REM **** THE VARIABLE NAMED ****
0C40 150C REM **** 'CK.5010$' ****
0C40 150C REM ****
0C40 150C REM **** OUTPUT : 'RT.5010' IS THE RETURN CODE ****
0C40 150C REM **** VARIABLE. IF THIS VARIABLE ****
0C40 150C REM **** CONTAINS ANY NUMBER OTHER ****
0C40 150C REM **** THAN 0, AN ERROR WAS FOUND ****
0C40 150C REM **** IN THE STRING. ****
0C40 150C REM ****
0C40 150C REM **** RESERVED LINE ****
0C40 150C REM **** NUMBERS : 5011-5019 ****
0C40 150C REM *****
0C40 150C RT.5010 = 0
0C47 150C
0C47 150C FOR I5010 = 1 TO LEN(CK.5010$)
0C57 150C J5010= ASC(MID$(CK.5010$,I5010,1))
0C68 1512 IF J5010 < 48 OR J5010 > 57 THEN RT.5010 = RT.5010 + 1
0C93 1512 NEXT I5010
0CA4 1512
0CA4 1512 RETURN
0CA7 1512 REM ----- END OF SUBROUTINE 5010 -----
0CA7 1512 5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTPATIENT UCA CHECK SUB
0CA8 1512 5700 REM $INCLUDE: 'RACS5700.SUB' INCLUDE THE ARRAY CONVERTER
0CA9 1512 REM *****
0CA9 1512 REM **** AMBULATORY CARE DATA BASE 29 JUL 85 ****
0CA9 1512 REM **** D R BOLLING ****
0CA9 1512 REM **** SUBROUTINE NAME : RXXS5700.SUB ****
0CA9 1512 REM **** SCANNER PROGRAM # : ALL ****
0CA9 1512 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0CA9 1512 REM **** CONVERTS A BINARY ARRAY INTO ****
0CA9 1512 REM **** TWO CHAR CODES. ****
0CA9 1512 REM ****
0CA9 1512 REM **** INPUT : INP.STOS AS STRING ****
0CA9 1512 REM ****
0CA9 1512 REM ****
0CA9 1512 REM **** OUTPUT : BUF.STOS AS STRING. ****
0CA9 1512 REM ****
0CA9 1512 REM ****
0CA9 1512 REM **** RESERVED LINE ****

```

```

Offset Data Source Line
OCA9 1512 REM **** NUMBERS : 5710-5730 ****
OCA9 1512 REM *****
OCA9 1512
OCA9 1512 BUF.STO$=""
OCB2 1512 N.STO=1
OCB9 1514 5710 X.STO=INSTR(N.STO,INP.STO$,"1")
OCCB 1516 IF X.STO=0 THEN GOTO 5720 'THATS ALL
OCDA 1516 N.STO = X.STO + 1 'NEXT STARTING POINT
OCE2 1516 X.STO = X.STO + 100 'PAD WITH LEADING ZERO
OCEC 1516 BUF.STO$ = BUF.STO$ + RIGHT$(STR$(X.STO),2)
OD02 1516 IF N.STO <= LEN(INP.STO$) THEN GOTO 5710
OD15 1516
OD15 1516 5720 RETURN
OD18 1516
OD18 1516 REM ----- END OF SUBROUTINE 5700 -----
OD18 1516
OD18 1516 6000 REM $INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE ROUTINE
OD19 1516 REM *****
OD19 1516 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
OD19 1516 REM **** SKIP COLE ****
OD19 1516 REM **** SUBROUTINE NAME : RXXS6000.SUB ****
OD19 1516 REM **** SCANNER PROGRAM # : ALL ****
OD19 1516 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
OD19 1516 REM **** PERFORMS INSTRING SEARCH ****
OD19 1516 REM ****
OD19 1516 REM **** INPUT : STRING TO BE SEARCHED MUST ****
OD19 1516 REM **** BE IN THE VARIABLE NAMED : ****
OD19 1516 REM **** 'X$' ****
OD19 1516 REM ****
OD19 1516 REM **** OUTPUT : 'TOT' = TOTAL NUMBER OF ****
OD19 1516 REM **** HITS IN THE DESTSTRING ****
OD19 1516 REM **** 'HOLD$( )' IS THE ARRAY ****
OD19 1516 REM **** CONTAINING THE NUMERIC ****
OD19 1516 REM **** VALUE OF THE HIT POSITIONS ****
OD19 1516 REM ****
OD19 1516 REM *****
OD19 1516 REM **** RESERVED LINE ****
OD19 1516 REM **** NUMBERS : 6001-6009 ****
OD19 1516 REM *****
OD19 1516 6001 PTR = INSTR(X$, "1")
OD27 1516 TOT = 0
OD2E 1516 WHILE PTR > 0
OD39 1516 TOT=TOT+1
OD41 1516 HOLD$(TOT) = RIGHT$(STR$(PTR),2)
OD43 1516 PTR=PTR+1
OD68 1516 PTR = INSTR(PTR,X$,"1")
OD7D 1516 WEND
OD81 1516 RETURN
OD84 1516
OD84 1516 REM -----END OF SUBROUTINE RXXS6000.SUB-----
OD84 1516
OD84 1516
OD84 1516 7000 REM $INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
OD85 1516 REM *****

```

Offset	Data	Source Line
0085	1516	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0085	1516	REM **** SKIP COLE ****
0085	1516	REM **** SUBROUTINE NAME : RACS7000.SUB ****
0085	1516	REM **** SCANNER PROGRAM # : ALL ****
0085	1516	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0085	1516	REM **** PRINTS THE STANDARD SCREEN ****
0085	1516	REM **** HEADING. ****
0085	1516	REM **** INPUT : COMMON VARIABLE USERS(2) ****
0085	1516	REM **** SYSTEM DATE ****
0085	1516	REM **** ****
0085	1516	REM **** OUTPUT : SCREEN HEADING ****
0085	1516	REM **** ****
0085	1516	REM **** RESERVED LINE ****
0085	1516	REM **** NUMBERS : 7001-7010 ****
0085	1516	REM ****
0085	1516	7001 LOCATE 1,1
008F	1516	PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
0097	1516	LOCATE 1,65
00A4	1516	PRINT DATES;
00AC	1516	LOCATE 2,1
00B9	1516	PRINT "USER : ";USERS(1)
00C6	1516	RETURN
00C9	1516	7100 REM \$INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
00CA	1516	REM ****
00CA	1516	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
00CA	1516	REM **** SKIP COLE ****
00CA	1516	REM **** SUBROUTINE NAME : RXXS7100.SUB ****
00CA	1516	REM **** SCANNER PROGRAM # : ALL ****
00CA	1516	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
00CA	1516	REM **** PRINTS THE STANDARD HEADING ****
00CA	1516	REM **** ON THE PRINTER. ****
00CA	1516	REM **** INPUT : DATE,PAGE,PGMIDS,PGMTITLS ****
00CA	1516	REM **** ****
00CA	1516	REM **** OUTPUT : PRINTER HEADING, LN.COUNT ****
00CA	1516	REM **** ****
00CA	1516	REM **** RESERVED LINE ****
00CA	1516	REM **** NUMBERS : 7101-7110 ****
00CA	1516	REM ****
00CA	1516	7101 IF PAGE > 0 THEN LPRINT CHR\$(12);
00E0	1516	LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITLS;
00ED	1516	LPRINT TAB(70);DATES
00F0	1516	PAGE=PAGE+1
00F8	1516	LPRINT "PROGRAM ";PGMIDS;TAB(70);"PAGE";
00E5	1516	LPRINT USING "####";PAGE
00F1	1516	LPRINT
00F9	1516	LN.COUNT=3
00E4	1516	RETURN
00E3	1516	
00E3	1516	8000 REM \$INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
00E4	1516	REM ****
00E4	1516	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ***
00E4	1516	REM **** SKIP COLE ***

```

Offset  Data      Source Line
-----  -
0E44 1516 REM ***** SUBROUTINE NAME      :  RXXS8000.SUB          ***
0E44 1516 REM ***** SCANNER PROGRAM #   :  ALL                  ***
0E44 1516 REM ***** FUNCTION           :  THIS SUBROUTINE MODULE  ***
0E44 1516 REM *****                  IS A GROUPING THAT PERFORMS ***
0E44 1516 REM *****                  VARIOUS DECODING FUNCTIONS ***
0E44 1516 REM *****                  ON THE SCANNER DATA     ***
0E44 1516 REM *****                  ***
0E44 1516 REM ***** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
0E44 1516 REM ***** 8050 - CHECK FOR END OF JOB                 ***
0E44 1516 REM ***** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
0E44 1516 REM ***** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
0E44 1516 REM *****                  (RETURNED IN TEXT$ STRING VARIABLE) ***
0E44 1516 REM *****                  ***
0E44 1516 REM ***** INPUT              :  SHEET RECORD, RECORD LENGTH ***
0E44 1516 REM *****                  ***
0E44 1516 REM ***** OUTPUT            :  'TEXT$' TRING VARIABLE ***
0E44 1516 REM *****                  ***
0E44 1516 REM ***** RESERVED LINE ***
0E44 1516 REM ***** NUMBERS              :  8001-8500          ***
0E44 1516 REM ***** *****
0E44 1516
0E44 1516 'DECODE THE HEADER ONLY
0E44 1516 8001    POINTER = 0
0E48 1516        RECORDPTR = VARPTR(SHEETREC(0))
0E52 1518        FOR J8000 = 1 TO 21
0E59 1518 8002    TEXT$= TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
0E77 1518        POINTER=POINTER+1
0E7F 1518        NEXT J8000
0E8E 151A        PROGRAM$= LEFT$(TEXT$,3)
0E9D 151E        BATCH$=  MID$(TEXT$,4,3)
0EAF 1522        SERIAL$= MID$(TEXT$,7,4)
0EC1 1526        RUNID$=  MID$(TEXT$,11,1)
0ED3 152A        FORM$=   MID$(TEXT$,12,2)
0EE5 152E        POCKET$= MID$(TEXT$,14,1)
0EF7 1532        SCANERR1$=MID$(TEXT$,16,2)
0F09 1536        SCANERR2$=MID$(TEXT$,18,2)
0F1B 153A        SCANERR3$=MID$(TEXT$,20,2)
0F2D 153E        GOTO 8500
0F31 153E
0F31 153E 8050 REM CHECK FOR END OF JOB/END OF BATCH
0F32 153E        IF PROGRAM$ = PG MID$ THEN GOTO 8500
0F44 153E        LPRINT STRINGS(80,"")
0F52 153E        LPRINT
0F5A 153E        LPRINT "RECORDS PROCESSED ... ";SERIAL$
0F67 153E        LPRINT "STARTED AT ..... ";BTIM$
0F74 153E        LPRINT "ENDED   AT ..... ";TIM$
0F81 153E        LPRINT CHR$(12)
0F8C 153E        GOTO 30000
0F90 153E
0F90 153E 8070 REM CHECK FOR SCANNER ERRORS
0F91 153E        IF POCKET$ = " " GOTO 8500
0FA3 153E        LPRINT LITHOS;
0FA3 153E        LPRINT " ... SCANNER ERRORS : ";
0FB3 153E        LPRINT SCANERR1$;" / ";

```

Offset	Data	Source Line
0FC0	153E	LPRINT SCANNER2\$;" / ";
0FC0	153E	LPRINT SCANNER3\$
0FD5	153E	LN=LN+1
0FDD	1540	GOTO 999
0FE1	1540	
0FE1	1540	8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
0FE2	1540	LOCATE 5,1:PRINT "PROGRAM ";PROGRAM\$;
0FFC	1540	
0FFC	1540	PRINT " BATCH ";BATCH\$;
1009	1540	PRINT " RUN ";RUNID\$;
1016	1540	PRINT " FORM ";FORM\$;
1023	1540	PRINT " POCKET ";POCKET\$
1030	1540	GOTO 8500
1034	1540	
1034	1540	8200 REM DECODE THE RESPONSE POSITIONS
1035	1540	POINTER = 21
103C	1540	RECORDPTR = VARPTR(SHEETREC(0))
1043	1540	FOR J8000 = 22 TO RECORDLENGTH
1050	1542	8202 TEXT\$ = TEXT\$+CHR\$(PEEK(RECORDPTR + POINTER))
106E	1542	POINTER=POINTER+1
1076	1542	NEXT J8000
1087	1542	
1087	1542	8500 RETURN
108A	1542	
108A	1542	REM ----- END OF RXXS8000.SUB -----
108A	1542	
108A	1542	9000 REM \$INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
108B	1542	REM *****
108B	1542	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
108B	1542	REM **** SKIP COLE ****
108B	1542	REM **** PROGRAM NAME : RACS9000.SUB ****
108B	1542	REM **** SCANNER PROGRAM # : ALL ****
108B	1542	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
108B	1542	REM **** CONTROLS THE SCANNER I/O ****
108B	1542	REM ****
108B	1542	REM **** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS ****
108B	1542	REM **** COMMUNICATIONS MANUAL AND THE ****
108B	1542	REM **** PRE-RELEASED SOFTWARE GUIDE ****
108B	1542	REM ****
108B	1542	REM *****
108B	1542	REM **** RESERVED LINE ****
108B	1542	REM **** NUMBERS : 9001-9100 ****
108B	1542	REM *****
108B	1542	
108B	1542	REM *****
108B	1542	REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
108B	1542	REM **** ARGUMENTS: PRESET ... SEE BELOW ****
108B	1542	REM *****
108B	1542	9001 REM
108C	1542	PROTOCOL(0) = 9600 'BAUD RATE
1093	1542	PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
109A	1542	PROTOCOL(2) = 8 'DATA BITS
10A1	1542	PROTOCOL(3) = 1 'STOP BITS

Offset	Data	Source Line
10A8	1542	PROTOCOL(4) = 2 'RS-232 PORT
10AF	1542	PROTOCOL(5) = 0 'WRITE TIME-OUT
10B6	1542	PROTOCOL(6) = 0 'READ TIME-OUT
10BD	1542	
10BD	1542	ERRSTAT\$ = SPACES(60)
10C9	1542	ARGPTR = VARPTR(PROTOCOL(0))
10D0	1544	CALL SETUP (ARGPTR,ERRSTAT\$)
10E1	1544	ERRMSG\$=""
10EA	1544	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="SETUP ERROR "+ERRSTAT\$
1106	1544	GOTO 9100
110A	1544	
110A	1544	REM *****
110A	1544	REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
110A	1544	REM **** ARGUMENTS: CNTRLOPT ****
110A	1544	REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
110A	1544	REM **** CNTRLOPT = 2 = STOP SCANNER (SO) ****
110A	1544	REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
110A	1544	REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
110A	1544	REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
110A	1544	REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
110A	1544	REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
110A	1544	REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
110A	1544	REM *****
110A	1544	9010 REM
110B	1544	ERRSTAT\$ = SPACES(60)
1117	1544	CALL CNTRL (CNTRLOPT,ERRSTAT\$)
1128	1544	ERRMSG\$=""
1131	1544	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
114D	1544	GOTO 9100
1151	1544	
1151	1544	REM *****
1151	1544	REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
1151	1544	REM ****
1151	1544	REM **** ARGUMENTS: READTYPE ****
1151	1544	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
1151	1544	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
1151	1544	REM ****
1151	1544	REM **** ARGUMENTS: RECORDLENGTH ****
1151	1544	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
1151	1544	REM **** TRANSMITTED ****
1151	1544	REM *****
1151	1544	9020 REM
1152	1544	ERRSTAT\$ = SPACES(60)
115E	1544	RECORDPTR = VARPTR(SHEETREC(0))
1165	1544	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
117E	1544	ERRMSG\$=""
1187	1544	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
11A8	1544	GOTO 9100
11AC	1544	
11AC	1544	REM *****
11AC	1544	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
11AC	1544	REM ****
11AC	1544	REM **** ARGUMENTS: PRINTPOS ****
11AC	1544	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****

RACP860

IMMUNIZATION SHORT FORM

PAGE 18

04-24-97

12:51:22

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
11AC	1544	REM **** VALUES = 0 THRU 90 ****
11AC	1544	REM ****
11AC	1544	REM **** ARGUMENTS: PSTRINGS ****
11AC	1544	REM **** TEXT TO BE PRINTED ON THE FORM ****
11AC	1544	REM ****
11AC	1544	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
11AC	1544	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
11AC	1544	REM ****
11AC	1544	9030 REM
11AD	1544	ERRSTAT\$ = SPACES(60)
11B9	1544	RECORDPTR = VARPTR(SHEETREC(0))
11C0	1544	CALL TPRINT(PRINTPOS,PSTRINGS,ERRSTAT\$)
11D5	154A	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
11F1	154A	GOTO 9100
11F5	154A	
11F5	154A	9100 RETURN
11F8	154A	REM -----END OF SUBROUTINE RACS9000.SUB -----
11F8	154A	
11F8	154A	REM END OF SUBROUTINES =====
11F8	154A	
11F8	154A	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
11F9	154A	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1201	154A	LPRINT "ERASING FILE ";DATFIL\$
120E	154A	BEEP
1212	154A	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
121E	154A	CLOSE
1222	154A	OPEN DATFIL\$ FOR OUTPUT AS #1
1234	154A	PRINT #1,STRINGS(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
1246	154A	CLOSE
124A	154A	
124A	154A	30000 REM
124B	154A	CLOSE
124F	154A	CHAIN "RACP10"
1256	154A	END
125A	154A	
125D	154A	

22151 Bytes Available

16981 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

PAGE 1

07-06-87

15:02:00

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

001A 0002 REM \$LINESIZE: 132

001A 0002 REM \$PAGESIZE: 66

001A 0002 REM \$TITLE: 'RACP900 '

001A 0002 REM \$SUBTITLE: 'OCC HEALTH FORM DESTSTRING/DECODE PROGRAM'

001A 0002 REM \$PAGE

RACP900

OCC HEALTH FORM DESTING/DECODE PROGRAM

PAGE 2

07-06 87

15:02:00

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```

001A 0002 REM +-----+
001A 0002 REM | NAME: RACP900          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | DATE: 02 APR 87      OCCUPATIONAL HEALTH FORM PROGRAM |
001A 0002 REM | D R BOLLING                               |
001A 0002 REM +-----+
001A 0002 REM          OCC HEALTH FORM OMR INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the base form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RACP05, otherwise,
001A 0002 REM the program chains to program RACP10 on exit.
001A 0002
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD          DIMENSION DEFINITIONS *
001A 0002 '* Date: 28 Feb 84          Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USERS(2),MOLENGTH(12),DATEERRS(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002
001A 0002 DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 DIM YNS(3) '(YES/NO ANSWERS 0=?, 1 = Y , 2=N , 3=X)
001A 0002 DIM ED.MSGS(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002 DIM CLINIC1.PFX$(5) '(PREFIX -B D F G S- FOR CLINIC #1)
001A 0002 DIM CLINIC2.PFX$(6) '(PREFIX -A B C D F S- FOR CLINIC #2)
001A 0002 DIM PROVIDER.TIMES(22) '(TIME TABLE FOR PROVIDERS)
001A 0002 DIM PROCED$(125) '(PROCEDURE TABLE FOR BASE FORM)
001A 0002 DIM DIAGN.TAB$(225) '(DIAGNOSIS TABLE FOR BASE FORM)

```

```

Offset  Data  Source Line
001A 0002      DIM OCCHES(20)      '(TABLE FOR OTHER CODES)
001A 0002      DIM HOLDS(99)      '(HOLD AREA FOR SUBROUTINE 6000)
001A 0002      REM $INCLUDE: 'RACCMN.MOD'      REM INCLUDE THE COMMON AREA DEFINITION
001A 0002      *****
001A 0002      '* NAME: RACCMN.MOD      COMMON AREA DEFINITION      *
001A 0002      '* Date: 28 Feb 84      Written by: Floyd Cole      *
001A 0002      *****
001A 0002      ' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A 0002      ' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A 0002      '
001A 0002      ' This program segment may be modified, but all files containing
001A 0002      ' an include for this segment must be re*compiled in order to
001A 0002      ' affect the changes made here.
001A 0002      '
001A 0002      ' *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002      COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL$ 'BASIC SCREEN COLORS
001A 0002      COMMON HEADERS$      '21 CHARACTER SCANNER HEADER INFO
001A 0002      COMMON TEXT$      ' ' AIMING CHARACTERS FROM SCANNER
001A 0002      COMMON PGMIDS$      'PROGRAM OR FORM ID
001A 0002      COMMON MOLENGTH( )      'DAYS IN THE MONTH
001A 0002      COMMON USERS( )
001A 0002      ' *****END OF COMMON DEFINITION*****
001A 0002
001A 0002      REM $INCLUDE: 'RACDEF.MOD'      REM INCLUDE THE DEFAULT DEFINITIONS
001A 0002      *****
001A 0002      '* NAME: RACP01.DEF      DEFAULT DEFINITIONS      *
001A 0002      '* Date: 28 Feb 84      Written by: Floyd Cole      *
001A 0002      *****
001A 0002      ' Variables used in common that have a default value on start*up
001A 0002      ' will be held in this file. It is an included file so it cannot
001A 0002      ' be run in a stand*alone mode. In normal operation, this file
001A 0002      ' should be 'included' in the main program only (RACP10.BAS).
001A 0002      '
001A 0002      ' This program segment may be modified, but all files containing
001A 0002      ' an include for this segment must be re*compiled in order to
001A 0002      ' affect the changes made here.
001A 0002      '
001A 0002      ' *****START OF DEFAULT DEFINITION*****
001A 0002
001A 0002      FORE = 15      'FOREGROUND COLOR = INTENSE WHITE
004B 1650      BACK = 1      'Background Color = Light Blue
004F 1650      BORD = 4      'BORDER = RED
0056 1652      HIDE = 4      'ALTERNATE COLOR = RED
005D 1652      EFORE = 14      'ERROR FOREGROUND DISPLAY
0064 1652      EBACK = 0      'ERROR BACKGROUND DISPLAY
006B 1652      BELL$ = CHR$(7) 'Sound the bell
0077 1652
0077 1652      MOLENGTH(1) = 31      'JAN
007E 1652      MOLENGTH(2) = 28      'FEB ---MODIFIED IN SUBROUTINE RACS5000.SUB
0085 1652      MOLENGTH(3) = 31      'MAR
009C 1652      MOLENGTH(4) = 30      'APR

```

Offset Data Source Line

```

0093 1652      MOLENGTH(5) = 31      'MAY
009A 1652      MOLENGTH(6) = 30      'JUN
00A1 1652      MOLENGTH(7) = 31      'JUL
00A8 1652      MOLENGTH(8) = 31      'AUG
00AF 1652      MOLENGTH(9) = 30      'SEP
00B6 1652      MOLENGTH(10) = 31     'OCT
00BD 1652      MOLENGTH(11) = 30     'NOV
00C4 1652      MOLENGTH(12) = 31     'DEC
00CB 1652
00CB 1652      DATEERR$(0) = " "
00D4 1652      DATEERR$(1) = "INVALID MONTH"
00DD 1652      DATEERR$(2) = "INVALID DAY "
00E6 1652      DATEERR$(3) = "DAY TOO LARGE FOR MONTH CODED"
00EF 1652
00EF 1652      MAXLENGTH = 80      'MAXIMUM LENGTH OF OUTPUT RECORD
00F6 1654      PADS = ". "      'PAD CHARACTER FOR SHORT RECORDS
00FF 1658
00FF 1658      ' *****END OF DEFAULT DEFINITION*****
00FF 1658
00FF 1658      KEY OFF
0105 1658
0105 1658      REM *****
0105 1658
0105 1658      REM *** DEFAULT CLINIC CODE TABLE ***
0105 1658      CLIDEFS="FBGA"
010E 165C
010E 165C      REM *** CLINIC CODE BY SINGLE BUBBLE ***
010E 165C      CLIBUBS="BHDAFBGBHGA"      'OCC HEALTH
0117 1660
0117 1660      REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0117 1660      REM      CLINIC #1      CLINIC #2
0117 1660      CLINIC1.PFX$(0)=" "      : CLINIC2.PFX$(0)=" "
0129 1660      CLINIC1.PFX$(1)="B"      : CLINIC2.PFX$(1)="A"
013B 1660      CLINIC1.PFX$(2)="D"      : CLINIC2.PFX$(2)="B"
014D 1660      CLINIC1.PFX$(3)="F"      : CLINIC2.PFX$(3)="C"
015F 1660      CLINIC1.PFX$(4)="G"      : CLINIC2.PFX$(4)="D"
0171 1660      CLINIC1.PFX$(5)="S"      : CLINIC2.PFX$(5)="F"
0183 1660      CLINIC2.PFX$(6)="S"
018C 1660
018C 1660      REM *** ENCOUNTER FORM PROVIDER TIME TABLE ***
018C 1660
018C 1660      PROVIDER.TIMES$(00)="000"      ' NO TIME
0195 1660      PROVIDER.TIMES$(01)="002"      ' 2 MINUTES
019E 1660      PROVIDER.TIMES$(02)="005"      ' 5 MINUTES
01A7 1660      PROVIDER.TIMES$(03)="010"      ' 10 MINUTES
01B0 1660      PROVIDER.TIMES$(04)="015"      ' 15 MINUTES
01B9 1660      PROVIDER.TIMES$(05)="020"      ' 20 MINUTES
01C2 1660      PROVIDER.TIMES$(06)="030"      ' 30 MINUTES
01CB 1660      PROVIDER.TIMES$(07)="045"      ' 45 MINUTES
01D4 1660      PROVIDER.TIMES$(08)="060"      ' 1 HOUR
01DD 1660      PROVIDER.TIMES$(09)="090"      ' 1 HOURS/30 MINUTES
01E6 1660      PROVIDER.TIMES$(10)="120"      ' 2 HOURS
01EF 1660      PROVIDER.TIMES$(11)="150"      ' 2 HOURS/30 MINUTES

```

```

Offset  Data      Source Line
101F8  1660      PROVIDER.TIMES(12)="180"      ' 3 HOURS
10201  1660      PROVIDER.TIMES(13)="210"      ' 3 HOURS/30 MINUTES
1020A  1660      PROVIDER.TIMES(14)="240"      ' 4 HOURS
10213  1660      PROVIDER.TIMES(15)="270"      ' 4 HOURS/30 MINUTES
1021C  1660      PROVIDER.TIMES(16)="300"      ' 5 HOURS
10225  1660      PROVIDER.TIMES(17)="330"      ' 5 HOURS/30 MINUTES
1022E  1660      PROVIDER.TIMES(18)="360"      ' 6 HOURS
10237  1660      PROVIDER.TIMES(19)="390"      ' 6 HOURS/30 MINUTES
10240  1660      PROVIDER.TIMES(20)="420"      ' 7 HOURS
10249  1660      PROVIDER.TIMES(21)="450"      ' 7 HOURS/30 MINUTES
10252  1660      PROVIDER.TIMES(22)="480"      ' 8 HOURS
10258  1660
10258  1660      REM YES/NO TABLE
10258  1660          YNS(0)=" " : YNS(1)="Y" : YNS(2)="N" : YNS(3)="X"
1027F  1660
1027F  1660      REM *** TABLE OF OTHER CODES ***
1027F  1660          OCCHE$(01)="01002" : OCCHE$(11)="01012"
10291  1660          OCCHE$(02)="01003" : OCCHE$(12)="01013"
102A3  1660          OCCHE$(03)="01004" : OCCHE$(13)="01014"
102B5  1660          OCCHE$(04)="01005" : OCCHE$(14)="01015"
102C7  1660          OCCHE$(05)="01006" : OCCHE$(15)="01016"
102D9  1660          OCCHE$(06)="01007" : OCCHE$(16)="01017"
102EB  1660          OCCHE$(07)="01008" : OCCHE$(17)="01018"
102FD  1660          OCCHE$(08)="01009" : OCCHE$(18)="01019"
1030F  1660          OCCHE$(09)="01010" : OCCHE$(19)="01020"
10321  1660          OCCHE$(10)="01011" : OCCHE$(20)="01021"
10333  1660
10333  1660      REM *****
10333  1660          PNUM=VAL(PGMID$)/10
10345  1662      REM LENGTH OF STRING RECEIVED FROM THE OMR....
10345  1662          HEADER = 21
1034C  1664          RESPONSE= 513
10353  1666          RECORDLENGTH = HEADER + RESPONSE
1035E  1668
1035E  1668          N.PROC = 81      ' NUMBER OF PROCEDURES FOR THIS FORM
10365  166A          N.DIAG.COL=4      ' NUMBER OF DX COLUMNS ON THIS FORM
1036C  166C
1036C  166C          DATFIL$ = "VISIT.DAT"      'FILE TO BE INPUT TO FOCUS
10375  1670          BTIME$=TIME$      'SCAN START TIME
1037E  1674
1037E  1674      REM *** ENCOUNTER FORM PROCEDURE TABLE ***
1037E  1674          F.NAME$="RACPROC." + PGMID$
1038C  1678          OPEN F.NAME$ FOR INPUT AS #3
1039D  1678          FOR I600=0 TO 125
103A3  1678              INPUT #3,PROCD$(I600)
103BB  167A              IF PROCD$(I600)="ZZZZZ" THEN GOTO 4
103D7  167A              NEXT I600
103E6  167A      4      CLOSE #3
103ED  167A
103ED  167A      REM *** ENCOUNTER FORM DIAGNOSIS TABLE ***
103ED  167A          F.NAME$="RACDIAG." + PGMID$
103FB  167A          OPEN F.NAME$ FOR INPUT AS #3
1040C  167A          FOR I600 = 0 TO 225

```

RACP900

OCC HEALTH FORM DESTSTRING/DECODE PROGRAM

PAGE 4

07-06-87

15:02:00

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```
0412 167A          INPUT #3,DIAGN.TABS(1600)
042A 167A          IF DIAGN.TABS(1600)="ZZZZZ" THEN GOTO 6
0446 167A          NEXT 1600
0456 167A  6      CLOSE #3
045D 167A
045D 167A  REM  INCLUDE: 'UCACAMP.OPT'  INCLUDE THE OUTP UCA VALIDATE TABLE
045D 167A  REM  INCLUDE: 'UCACAMP.IPT'  INCLUDE THE INP UCA VALIDATE TABLE
045D 167A
045D 167A  REM $PAGE
```

Offset	Data	Source Line
045D	167A	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
0462	167A	CLS
0466	167A	GOSUB 7000 'PRINT SCREEN HEADING
0468	167A	
0468	167A	REM *****
0468	167A	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
0468	167A	REM *****
0468	167A	REM
0468	167A	OPEN DATFILS FOR APPEND AS #1
047D	167A	
047D	167A	REM *****
047D	167A	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
047D	167A	REM *****
047D	167A	LPRINT CHR\$(15);
0488	167A	WIDTH "LPT1:",160
0492	167A	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
049E	167C	COLOR 14
04A5	167C	LOCATE 11,26 : PRINT "BASE ENCOUNTER FORM "
048A	167C	COLOR FORE,BACK,BORD
04D0	167C	
04D0	167C	REM *****
04D0	167C	REM **** COMMUNICATIONS SETUP ****
04D0	167C	REM *****
04D0	167C	REM PROTOCOL
04D0	167C	GOSUB 9001
04D5	167C	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
04EF	1680	
04EF	1680	REM START SCANNER (S1)
04EF	1680	CNTRLOPT = 1 :GOSUB 9010
04FB	1682	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0515	1682	
0515	1682	LOCATE 22,25;PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
052A	1682	READTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
0531	1684	
0531	1684	REM *****
0531	1684	REM *** SET SCAN SHEET CALL ***
0531	1684	REM *****
0531	1684	REM
0531	1684	
0531	1684	10 REM - RETURN POINT TO READ NEXT SHEET
0532	1684	
0532	1684	AS=INKEY\$
053B	1688	IF AS=CHR\$(27) THEN GOTO 25000
0551	1688	
0551	1688	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
0556	1688	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
0572	168C	
0572	168C	TEXT\$="" 'CLEAR THE INPUT AREA
057B	168C	GOSUB 8000 'DECODE HEADER
0580	168C	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
0585	168C	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
058A	168C	LITHO\$ = MID\$(TEXT\$,22,8)
059C	1690	GOSUB 8070 'CHECK FOR SCANNER ERRORS
05A1	1690	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

Offset	Data	Source Line
05A6	1690	
05A6	1690	REM \$INCLUDE: 'RACH900.M01' INCLUDE THE READ FORM REFORMAT/EDIT MOD
05A6	1690	REM *****
05A6	1690	REM **** AMBULATORY CARE INFORMATION SYSTEM 20 MAY 87 ****
05A6	1690	REM **** D R BOLLING ****
05A6	1690	REM **** MODULE NAME : RACH900.M01 ****
05A6	1690	REM **** SCANNER PROGRAM # : GENERAL FORM ****
05A6	1690	REM ****
05A6	1690	REM **** PURPOSE : REFORMAT/EDIT THE FORM ****
05A6	1690	REM ****
05A6	1690	REM *****
05A6	1690	REM **** RESERVED LINE NUMBERS 100-199 ****
05A6	1690	REM *****
05A6	1690	
05A6	1690	N.ERR =0 'COUNTS THE NUMBER OF ERRORS
05AD	1692	
05AD	1692	REM *** LITHO CODE DONE IN BAS PROGRAM ***
05AD	1692	
05AD	1692	REM *** CLINIC ID (PREFIX + CODE) ***
05AD	1692	100 CL1.CODS=CLIDEFS 'DEFAULT CLINIC CODE
05B6	1696	
05B6	1696	REM ... CLINIC CODE WHEN SINGLE BUBBLED ...
05B6	1696	IF MID\$(TEXT\$,34,1)=" " THEN 102
05CE	1696	X=VAL(MID\$(TEXT\$,34,1))
05E4	1698	X=(X-1)*4+1 'CALCULATE OFFSET
05F4	1698	CL1.CODS=MID\$(CLIBUBS,X,4)
0607	1698	GOTO 104
060B	1698	
060B	1698	REM ... CLINIC CODE WHEN HAND CODED ...
060B	1698	102 IF MID\$(TEXT\$,30,4)=" " THEN 104
0623	1698	CK.X=VAL(MID\$(TEXT\$,30,1))
0639	169A	CK.CODS=MID\$(TEXT\$,31,3)
064B	169E	REM GOSUB 5500 'OUTP UCA CODE CHECK
064B	169E	REM IF RT.5500 = 0 THEN GOTO 102
064B	169E	REM N.ERR = N.ERR + 1
064B	169E	REM ED.MSG\$(N.ERR)="INVALID CLINIC CODE"
064B	169E	
064B	169E	CL1.CODS=CLINIC1.PFX\$(CK.X) + CK.CODS 'HAND CODED
0663	169E	
0663	169E	REM VISIT DATE
0663	169E	104 CMS=MID\$(DATES,1,2) 'CURRENT MONTH
0675	16A2	YRS=MID\$(DATES,9,2) 'CURRENT YEAR
0687	16A6	X\$=MID\$(TEXT\$,35,4) 'MONTH AND DAY
0699	16AA	IF LEFT\$(X\$,2)<=CMS THEN 105 'OK, USE THIS YEAR
06AE	16AA	YRS=RIGHT\$(STR\$(VAL(YRS)-1),2) 'USE LAST YEAR
06CE	16AA	105 VIDATES=YRS+X\$
06DC	16AE	D1\$=MID\$(X\$,3,1)
06EE	16B2	D2\$=MID\$(X\$,4,1)
0700	16B6	IF D1\$=" " AND D2\$<>" " THEN RT.5000=2 : GOTO 106
072E	16B8	IF D1\$<>" " AND D2\$=" " THEN RT.5000=2 : GOTO 106
075C	16B8	'EDIT VISIT DATE
075C	16B8	CK.5000\$=VIDATES
0765	16B8	GOSUB 5000 'DATE CHECK
076A	16B8	IF RT.5000=0 THEN GOTO 107

Offset Data Source Line

```

0779 168C 106 N.ERR=N.ERR+1
0781 168C ED.MSG$(N.ERR)="VISIT DATE" + DATEERR$(RT.5000)
07A7 168C
07A7 168C REM *** PRIMARY PROVIDER ***
07A7 168C 107 PPROV.PFX$=MID$(TEXT$,39,1)
07B9 16C0 PPROV.NUM$=MID$(TEXT$,40,4)
07CB 16C4
07CB 16C4 REM *** PATIENT SSN ***
07CB 16C4 108 SSN$ = MID$(TEXT$,44,9)
07DD 16C8
07DD 16C8 REM *** FAMILY MEMBER PREF ***
07DD 16C8 110 FHEMP$=MID$(TEXT$,53,2)
07EF 16CC
07EF 16CC REM *** FORM NUMBER ***
07EF 16CC FRM$=LEFT$(PGMID$,2)
07FE 16D0
07FE 16D0 REM *** VISIT COUNT ***
07FE 16D0 VCNT$="1" 'DEFAULT
0807 16D4
0807 16D4 REM *** JOB REL IF YES ***
0807 16D4 JOBRIF$=MID$(TEXT$,55,1)
0819 16D8 JOBRIF$="N"
0822 16D8 IF MID$(TEXT$,55,1)="1" THEN JOBRIF$="1"
0843 16D8 IF MID$(TEXT$,55,1)="2" THEN JOBRIF$="2"
0864 16D8 IF MID$(TEXT$,55,1)="3" THEN JOBRIF$="3"
0885 16D8
0885 16D8 REM *** PRIMARY PROVIDER TIME ***
0885 16D8 114 X=VAL(MID$(TEXT$,56,2))
089B 16D8 PPROV.TIM$=PROVIDER.TIMES(X)
08AD 16DC
08AD 16DC REM *** SECONDARY PROVIDER ***
08AD 16DC 115 SPROV.PFX$=MID$(TEXT$,58,1)
08BF 16E0 SPROV.NUM$=MID$(TEXT$,59,4)
08D1 16E4 IF SPROV.NUM$="" THEN PB=1 ELSE PB=0 'PROV 2 BLANK INDICATOR
08F0 16E6 IF SPROV.PFX$="" AND SPROV.NUM$<>"" THEN 116
0916 16E6 IF SPROV.PFX$<>"" AND SPROV.NUM$="" THEN 116
093C 16E6 GOTO 117
0940 16E6
0940 16E6 116 N.ERR = N.ERR +1
0948 16E6 ED.MSG$(N.ERR)="PROV 2 CODE MISSING PREFIX OR NUMBER"
095C 16E6
095C 16E6 REM *** SECONDARY PROVIDER TIME ***
095C 16E6 117 X=VAL(MID$(TEXT$,63,2))
0972 16E6 SPROV.TIM$=PROVIDER.TIMES(X)
0984 16EA REM IS THERE A TIME AND NO SEC PROV CODED?
0984 16EA IF SPROV.TIM$<>"000" AND PB=1 THEN 118
09A7 16EA REM IS THERE NO TIME AND A SEC PROV CODED?
09A7 16EA IF SPROV.TIM$="000" AND PB=0 THEN 119
09CA 16EA GOTO 120
09CE 16EA
09CE 16EA 118 N.ERR = N.ERR +1
09D6 16EA ED.MSG$(N.ERR)="TIME CODED WITH NO SEC PROV CODED"
09EA 16EA GOTO 120
09EE 16EA

```

Offset	Data	Source Line
09EE	16EA	119 N.ERR = N.ERR +1
09F6	16EA	ED.MSG\$(N.ERR)="NO PROV 2 TIME"
0A0A	16EA	
0A0A	16EA	REM *** REASON FOR SECONDARY PROVIDER ***
0A0A	16EA	120 SPROV.REAS=MID\$(TEXT\$,65,1)
0A1C	16EE	REM IS THERE A REASON AND NO SEC PROV CODED?
0A1C	16EE	IF SPROV.REAS<>" " AND PB=1 THEN 121
0A3F	16EE	REM IS THERE NO REASON AND A SEC PROV CODED?
0A3F	16EE	IF SPROV.REAS=" " AND PB=0 THEN 122
0A62	16EE	GOTO 123
0A66	16EE	
0A66	16EE	121 N.ERR = N.ERR +1
0A6E	16EE	ED.MSG\$(N.ERR)="REASON CODED WITH NO SEC PROV CODED"
0A82	16EE	GOTO 123
0A86	16EE	
0A86	16EE	122 N.ERR = N.ERR +1
0A8E	16EE	ED.MSG\$(N.ERR)="NO PROV 2 REASON"
0AA2	16EE	
0AA2	16EE	REM *** IF NOT SCHEDULED ***
0AA2	16EE	123 X=VAL(MID\$(TEXT\$,66,1))
0AB8	16EE	NOTSCS=RIGHT\$(STR\$(X+1),1) 'DEFAULT IS 1
0ACC	16F2	
0ACC	16F2	REM *** IF NOT CLINIC ***
0ACC	16F2	124 X=VAL(MID\$(TEXT\$,67,1))
0AE2	16F2	NOTCLS=RIGHT\$(STR\$(X+1),1) 'DEFAULT IS 1
0AF6	16F6	
0AF6	16F6	REM *** REFERRAL CODE ***
0AF6	16F6	REF.PFX\$=" "
0AFF	16FA	REF.COD\$=" "
0B08	16FE	
0B08	16FE	REM *** ILLNESS ***
0B08	16FE	126 ILLNS=MID\$(TEXT\$,68,1)
0B1A	1702	IF ILLNS="1" THEN ILLNS="N"
0B31	1702	IF ILLNS="2" THEN ILLNS="R"
0B48	1702	
0B48	1702	REM *** INJURY ***
0B48	1702	128 INJUS=MID\$(TEXT\$,69,1)
0B5A	1706	IF INJUS="1" THEN INJUS="N"
0B71	1706	IF INJUS="2" THEN INJUS="R"
0B88	1706	
0B88	1706	REM *** PURPOSE OF VISIT ***
0B88	1706	PURP\$=MID\$(TEXT\$,70,1)
0B9A	170A	REM CONVERT ALPH INTO NUM
0B9A	170A	PURPV=ASC(PURP\$)-64
0BA7	170C	IF PURP\$=" " THEN PURPV=0
0BBC	170C	
0BBC	170C	REM *** JOB RELATED VISIT ***
0BBC	170C	130 RELAT.VISS="N"
0BC5	1710	
0BC5	1710	REM *** MIL ONLY DUTY ***
0BC5	1710	132 MILDUTS=MID\$(TEXT\$,72,1)
0BD7	1714	
0BD7	1714	REM *** MIL ONLY QTRS ***
0BD7	1714	134 MILQTRS=MID\$(TEXT\$,73,1)

Offset Data Source Line

```

0BE9 1718
0BE9 1718 REM *** MIL ONLY PROF ***
0BE9 1718 136 MILPROS=MID$(TEXT$,74,1)
0BF8 171C
0BF8 171C REM *** NOT AVAILABLE ***
0BF8 171C 138 NAVAILS=MID$(TEXT$,75,1)
0C00 1720
0C00 1720 REM *** SPEC PREASSIGNED CLINIC ***
0C00 1720 140 INP.STOS=MID$(TEXT$,80,12) 'INCLUDE NOT AVAILABLE BUBBLES
0C1F 1724 GOSUB 5700 'CONVERT ARRAY
0C24 1724 SPE.BUFS=BUF.STOS 'UP TO 9 TWO DIGIT CODES
0C2D 172C
0C2D 172C REM *** SPEC PROGRAMS ***
0C2D 172C REM *** READ AS ARRAY ***
0C2D 172C 142 SPPROGS=" "
0C36 1730
0C36 1730 REM *** PROVIDER 2 ADDL PROC 1 ***
0C36 1730 144 PR2PRC1$=""
0C3F 1734 IF MID$(TEXT$,92,1)="1" THEN PR2PRC1$="Y"
0C60 1734 IF MID$(TEXT$,92,1)="2" THEN PR2PRC1$="3"
0C81 1734
0C81 1734 REM *** ADDITIONAL PROCEDURE 1 ***
0C81 1734 146 ADDP1$=MID$(TEXT$,93,5)
0C93 1738 IF PR2PRC1$="Y" AND PB=1 THEN 147
0CB6 1738 GOTO 148
0C8A 1738
0C8A 1738 147 N.ERR = N.ERR + 1
0CC2 1738 ED.MSG$(N.ERR)="ADDL PROC CODED FOR PROV 2 BUT PROV 2 NOT CODED"
0CD6 1738
0CD6 1738 REM *** PROVIDER 2 ADDL PROC 2 ***
0CD6 1738 148 PR2PRC2$=""
0CDF 173C IF MID$(TEXT$,98,1)="1" THEN PR2PRC2$="Y"
0D00 173C IF MID$(TEXT$,98,1)="2" THEN PR2PRC2$="3"
0D21 173C
0D21 173C REM *** ADDITIONAL PROCEDURE 2 ****
0D21 173C 150 ADDP2$=MID$(TEXT$,99,5)
0C33 1740 IF PR2PRC2$="Y" AND PB=1 THEN 151
0D56 1740 GOTO 152
0D5A 1740
0D5A 1740 151 N.ERR = N.ERR + 1
0D62 1740 ED.MSG$(N.ERR)="ADDL PROC CODED FOR PROV 2 BUT PROV 2 NOT CODED"
0D76 1740
0D76 1740 REM *** ADMITTED ***
0D76 1740 152 ADMITS$=""
0D7F 1744 IF MID$(TEXT$,104,1)="1" THEN ADMITS$="Y"
0DA0 1744
0DA0 1744 REM *** UNLISTED PRIMARY DX ***
0DA0 1744 154 X=VAL(MID$(TEXT$,105,1))
0DB6 1744 IF X=0 THEN PRIMDX$=""
0DCA 1748 IF X=1 THEN PRIMDX$="V"
0DDE 1748 IF X=2 THEN PRIMDX$="S"
0DF2 1748 DXTMP$=MID$(TEXT$,106,5)
0E14 174C IF DXTMP$="" " THEN GOTO 158
0E16 174C

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

0E16 174C      REM REMOVE LEADING BLANKS
0E16 174C 156 IF LEFT$(DXTMP$,1)=" " THEN DXTMP$=RIGHT$(DXTMP$,4)+" ":GOTO 156
0E45 174C
0E45 174C 158 LASTCS=RIGHT$(DXTMP$,1) 'GET LAST CHAR
0E54 1750      IF LASTCS<>" " AND LASTCS<>"0" THEN PRIMDX$="" 'REMOVE V OR S
0E80 1750      PRIMDX$=LEFT$(PRIMDX$+DXTMP$,5) 'GET 5 CHAR ONLY
0E92 1750
0E92 1750 REM *** UNLISTED SECONDARY DX ***
0E92 1750 160 X=VAL(MID$(TEXT$,111,1))
0EA8 1750      IF X=0 THEN SECDX$=""
0EBC 1754      IF X=1 THEN SECDX$="V"
0ED0 1754      IF X=2 THEN SECDX$="S"
0EE4 1754      DXTMP$=MID$(TEXT$,112,5)
0EF6 1754      IF DXTMP$=" " THEN GOTO 164
0F08 1754
0F08 1754 REM REMOVE LEADING BLANKS
0F08 1754 162 IF LEFT$(DXTMP$,1)=" " THEN DXTMP$=RIGHT$(DXTMP$,4)+" ":GOTO 162
0F37 1754
0F37 1754 164 LASTCS=RIGHT$(DXTMP$,1) 'GET LAST CHAR
0F46 1754      IF LASTCS<>" " AND LASTCS<>"0" THEN SECDX$="" 'REMOVE V OR S
0F72 1754      SECDX$=LEFT$(SECDX$+DXTMP$,5) 'GET 5 CHAR ONLY
0F84 1754
0F84 1754 REM *** FOLLOW UP/RULE OUT ***
0F84 1754 166 FU.RCS=MID$(TEXT$,117,1)
0F96 1758
0F96 1758 REM *** EVALUATION/SERV/PROC PROV 1 ***
0F96 1758 168 INP.STOS=MID$(TEXT$,118,N.PROC)
0FA9 1758      ESP.BUF1$=""
0FB2 175C      N.STO=1
0FB9 175E
0FB9 175E 169 X.STO=INSTR(N.STO,INP.STOS,"1")
0FCB 1760      IF X.STO=0 THEN GOTO 170 'THATS ALL
0FDA 1760      N.STO=X.STO + 1 'NEXT STARTING POINT
0FE2 1760      ESP.BUF1$=ESP.BUF1$ + PROCED$(X.STO) 'ADD CODE TO BUFFER BY FIVES
0FF7 1760      IF N.STO <= LEN(INP.STOS) THEN GOTO 169
100A 1760
100A 1760 REM *** EVALUATION/SERV/PROC PROV 2 ***
100A 1760 170 INP.STOS=MID$(TEXT$,118+N.PROC,N.PROC)
1021 1760      ESP.BUF2$=""
102A 1764      N.STO=1
1031 1764
1031 1764 171 X.STO=INSTR(N.STO,INP.STOS,"1")
1043 1764      IF X.STO=0 THEN GOTO 174 'THATS ALL
1052 1764      IF PB=1 THEN 172
1050 1764      N.STO=X.STO + 1 'NEXT STARTING POINT
1065 1764      ESP.BUF2$=ESP.BUF2$ + PROCED$(X.STO) 'ADD CODE TO BUFFER BY FIVES
107A 1764      IF N.STO <= LEN(INP.STOS) THEN GOTO 171
108D 1764      GOTO 174
1091 1764
1091 1764 172 N.ERR = N.ERR + 1
1099 1764      ED.MSG$(N.ERR)="PROC CODED FOR PROV 2 BUT PROV 2 NOT CODED"
10AD 1764
10AD 1764 REM *** EVALUATION/SERV/PROC OUT ***
10AD 1764 174 INP.STOS=MID$(TEXT$,118+2*N.PROC,N.PROC)

```

Offset	Data	Source Line
10C9	1764	ESP.BUF3\$=""
1002	1768	N.STC=1
1009	1768	
1009	1768	175 X.STO=INSTR(N.STO,INP.STOS,"1")
10E8	1768	IF X.STO=0 THEN GOTO 176 'THATS ALL
10FA	1768	N.STO=X.STO + 1 'NEXT STARTING POINT
1102	1768	ESP.BUF3\$=ESP.BUF3\$ + PROCEED\$(X.STO) 'ADD CODE TO BUFFER BY FIVES
1117	1768	IF N.STO <= LEN(INP.STOS) THEN GOTO 175
112A	1768	
112A	1768	176 T.POS = 118 + 3*N.PROC
1138	176A	
1138	176A	OTHN = 20 'NUMBER OF OTHER CODES
113F	176C	
113F	176C	REM *** OTHER CODES ***
113F	176C	178 INP.STOS=MID\$(TEXT\$,T.POS,OTHN)
1153	176C	OTH.BUF\$=""
115C	1770	N.STO=1
1163	1770	
1163	1770	180 X.STO=INSTR(N.STO,INP.STOS,"1")
1175	1770	IF X.STO=0 THEN GOTO 182 'THATS ALL
1184	1770	N.STO=X.STO + 1 'NEXT STARTING POINT
118C	1770	OTH.BUF\$=OTH.BUF\$ + OCCHE\$(X.STO) 'ADD CODE TO BUFFER BY FIVES
11A1	1770	IF N.STO <= LEN(INP.STOS) THEN GOTO 180
1184	1770	
1184	1770	182 T.POS = T.POS + OTHN
118F	1770	184 REM
11C0	1770	
11C0	1770	REM *** PRIMARY DX ***
11C0	1770	REM * IF OTHER PRIM DX IS CODED THEN SKIP THIS SECTION *
11C0	1770	186 IF PRIMDX\$<>" " THEN GOTO 191
11D2	1770	X.POS=T.POS 'STARTING POSITION
11D9	1772	C1\$12=33 : C3\$12=39 'NO. OF ITEMS IN EACH COL
11E7	1776	C2\$12=36 : C4\$12=26
11F5	177A	GOSUB 5800 'GET POSITION
11FA	177A	IF X.FIN=0 THEN GOTO 191
1209	177C	IF RT.5800=0 THEN GOTO 190
1218	177E	N.ERR=N.ERR + 1
1220	177E	ED.MSG\$(N.ERR)="PRIMARY DX HAS MULTIPLE CODES"
1234	177E	GOTO 190
1238	177E	
1238	177E	190 PRIMDX\$=DIAGN.TAB\$(X.FIN)
124A	177E	
124A	177E	191 REM
124B	177E	IF PURPV>0 AND PRIMDX\$<>" " THEN 192
1270	177E	IF PURPV=0 AND PRIMDX\$="" THEN 193
1255	177E	IF PURPV=0 THEN 194
12A0	177E	PRIMDX\$="S00"+RIGHT\$(STR\$(PURPV+100),2) 'USE S00 PLUS LAST TWO
128B	177E	GOTO 194 'DIGITS OF PURPOSE INDEX
128F	177E	
128F	177E	192 N.ERR=N.ERR + 1
12C7	177E	ED.MSG\$(N.ERR)="PRIMARY DX INCLUDING PURPOSE HAS MULTIPLE CODES"
1208	177E	GOTO 194
12D5	177E	
12D5	177E	193 N.ERR=N.ERR + 1

```

Offset  Data    Source Line
12E7 177E      ED.MSG$(N.ERR)="PRIMARY DX OR PURPOSE OF VISIT NOT CODED"
12FB 177E
12FB 177E 194  T.POS = T.POS + 2 * N.DIAG.COL
130A 177E      MDX = 133
1311 1780
1311 1780 REM *** SECONDARY DX ***
1311 1780      IMP.STO=MID$(TEXT$,T.POS,MDX)
1325 1780      DX2.BUFS=""
132E 1784      N.STO=1
1335 1784
1335 1784 196  X.STO=INSTR(N.STO,IMP.STO$,"1")
1347 1784      IF X.STO=0 THEN GOTO 197      'THATS ALL
1356 1784      N.STO=X.STO + 1      'NEXT STARTING POINT
135E 1784      DX2.BUFS=DX2.BUFS + DIAGN.TAB$(X.STO) 'ADD CODE TO BUFFER BY FIVES
1373 1784      IF N.STO <= LEN(IMP.STO$) THEN GOTO 196
1386 1784 197  IF SECDX$=""      " THEN GOTO 198
1398 1784      DX2.BUFS=DX2.BUFS+SECDX$      'ADD OTHER SEC DX IF THERE
13A4 1784
13A4 1784 198  T.POS = T.POS + MDX
13AF 1784
13AF 1784 REM *** DISPOSITIONS ***
13AF 1784 REM *** READ AS ARRAY ***
13AF 1784      IMP.STO=MID$(TEXT$,T.POS,13)
13C2 1784
13C2 1784      GOSUB 5700      'CONVERT ARRAY
13C7 1784      DISPO$=BUF.STO$      'UP TO 13 TWO DIGIT CODES
13D0 1788
13D0 1788 REM -----END OF MODULE RACH900.MO1-----
13D0 1788
13D0 1788 IF N.ERR = 0 THEN GOTO 997
13DF 1788      LPRINT "LITHO # ";LITHO$;" ...ERRORS"
13F1 1788      FOR 1997 = 1 TO N.ERR
13FE 178A          LPRINT USING "### ";1997;
140A 178C          LPRINT "=> ";ED.MSG$(1997)
1420 178C      NEXT 1997
1431 178C      LN.COUNT = LN.COUNT + N.ERR + 1
143D 178E      CNTRLPT = 6
1444 178E      GOSUB 9010      'REJECT THE FORM
1449 178E      GOTO 998      'BYPASS THE DISK WRITER....
144D 178E
144D 178E 997 REM $INCLUDE: 'RACH900.MO2' REM INCLUDE THE BASE FORM DISK WRITER
144E 178E REM *****
144E 178E REM **** AMBULATORY CARE INFORMATION SYSTEM      3 APR 87      ****
144E 178E REM ****                                D R BOLLING      ****
144E 178E REM **** MODULE NAME      : RACH900.MO2      ****
144E 178E REM **** SCANNER PROGRAM # : OCC HEALTH FORM      ****
144E 178E REM ****                                ****
144E 178E REM **** PURPOSE      : CREATE AND WRITE THE DISK      ****
144E 178E REM ****                                RECORD FOR INPUT TO FOCUS      ****
144E 178E REM ****                                ****
144E 178E REM *****
144E 178E REM **** RESERVED LINE NUMBERS 200-299      ****
144E 178E REM *****
144E 178E

```

```

Offset Data Source Line
144E 178E REM BUILD THE OUTPUT RECORD
144E 178E
144E 178E GOSUB 270 'BUILD THE RECORD KEY
1453 178E
1453 178E REM *****
1453 178E REM **** RECORD TYPE "1" - MAIN TRANSACTION ****
1453 178E REM *****
1453 178E
1453 178E GOSUB 272 'BUILD DATA FOR TYPE 1
1458 178E RECOUNT$=PGMID$+"1"+RECKEY$+RECOD1$ 'TRANSACTION ID PLUS RECORD
1472 179A GOSUB 280
1477 179A PRINT #1,RECOUNT$
1482 179A
1482 179A REM *****
1482 179A REM **** RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE ****
1482 179A REM *****
1482 179A
1482 179A REM *** ADD ADDITIONAL PROCEDURES IF ANY ***
1482 179A IF ADP1$<>" " AND PR2PRC1$="N" THEN ESP.BUF1$=ESP.BUF1$+ADP1$
1484 179A IF ADP2$<>" " AND PR2PRC2$="N" THEN ESP.BUF1$=ESP.BUF1$+ADP2$
14E6 179A
14E6 179A IF ADP1$<>" " AND PR2PRC1$="Y" THEN ESP.BUF2$=ESP.BUF2$+ADP1$
1518 179A IF ADP2$<>" " AND PR2PRC2$="Y" THEN ESP.BUF2$=ESP.BUF2$+ADP2$
154A 179A
154A 179A IF ADP1$<>" " AND PR2PRC1$="3" THEN ESP.BUF3$=ESP.BUF3$+ADP1$
157C 179A IF ADP2$<>" " AND PR2PRC2$="3" THEN ESP.BUF3$=ESP.BUF3$+ADP2$
15AE 179A
15AE 179A REM *** PROCESS PROV 1 PROCEDURES ***
15AE 179A
15AE 179A IF LEN(ESP.BUF1$)=0 THEN GOTO 206
158F 179A
158F 179A RPOINT=1
15C6 179C 202 RECOD2$=MID$(ESP.BUF1$,RPOINT,5)
15D9 17A0 IF RECOD2$=" " THEN GOTO 204
15EB 17A0
15EB 17A0 RECOUNT$=PGMID$+"2"+RECKEY$+"1"+RECOD2$ 'TRANSACTION ID PLUS RECORD
1608 17A0 GOSUB 280
1610 17A0 PRINT #1,RECOUNT$
1618 17A0
1618 17A0 204 RPOINT = RPOINT + 5
1625 17A0 IF RPOINT < LEN(ESP.BUF1$) THEN GOTO 202
1638 17A0
1638 17A0 REM *** PROCESS PROV 2 PROCEDURES ***
1638 17A0
1638 17A0 206 IF LEN(ESP.BUF2$)=0 THEN GOTO 211
1649 17A0
1649 17A0 RPOINT=1
1650 17A0 208 RECOD2$=MID$(ESP.BUF2$,RPOINT,5)
1663 17A0 IF RECOD2$=" " THEN GOTO 210
1675 17A0
1675 17A0 RECOUNT$=PGMID$+"2"+RECKEY$+"2"+RECOD2$ 'TRANSACTION ID PLUS RECORD
1693 17A0 GOSUB 280
169C 17A0 PRINT #1,RECOUNT$
16A8 17A0

```

Offset	Data	Source Line
16A8	17A0	210 RPOINT = RPOINT + 5
16B2	17A0	IF RPOINT < LEN(ESP.BUF2\$) THEN GOTO 208
16C5	17A0	16C5 17A0 REM *** PROCESS OUT PROCEDURES ***
16C5	17A0	16C5 17A0
16C5	17A0	211 IF LEN(ESP.BUF3\$)=0 THEN GOTO 214
16D6	17A0	16D6 17A0
16D6	17A0	RPOINT=1
16D0	17A0	212 RECOD2\$=MID\$(ESP.BUF3\$,RPOINT,5)
16F0	17A0	IF RECOD2\$="" THEN GOTO 213
1702	17A0	1702 17A0
1702	17A0	RECOU2\$=PGMID\$+"2"+RECKEY\$+"3"+RECOD2\$ 'TRANSACTION ID PLUS RECORD
1722	17A0	GOSUB 280
1727	17A0	PRINT #1,RECOU2\$
1732	17A0	1732 17A0
1732	17A0	213 RPOINT = RPOINT + 5
173C	17A0	IF RPOINT < LEN(ESP.BUF3\$) THEN GOTO 212
174F	17A0	174F 17A0
174F	17A0	214 REM END OF TYPE 2 RECORDS
1750	17A0	1750 17A0
1750	17A0	REM *****
1750	17A0	REM **** RECORD TYPE "3" - RECKEY PLUS SPECIFIC PRE CLINIC CODES ****
1750	17A0	REM *****
1750	17A0	IF LEN(SPE.BUF\$)=0 THEN 219
1750	17A0	1750 17A0
1750	17A0	RPOINT=1
1764	17A0	215 X3\$=MID\$(SPE.BUF\$,RPOINT,2)
1777	17A4	IF X3\$="10" THEN RECOD3\$="M" :GOTO 216
1792	17A8	IF X3\$="11" THEN RECOD3\$="L" :GOTO 216
17AD	17A8	IF X3\$="12" THEN RECOD3\$="X" :GOTO 216
17C8	17A8	17C8 17A8
17C8	17A8	RECOD3\$=RIGHT\$(X3\$,1)
17D7	17A8	IF RECOD3\$="" THEN 218
17E5	17A8	17E5 17A8
17E5	17A8	216 RECOU3\$=PGMID\$ + "3" + RECKEY\$ + RECOD3\$
17FF	17A8	GOSUB 280
1804	17A8	PRINT #1,RECOU3\$
180F	17A8	180F 17A8
180F	17A8	218 RPOINT=RPOINT+2
1818	17A8	IF RPOINT < LEN(SPE.BUF\$) THEN 215
1827	17A8	1827 17A8
1827	17A8	219 REM END OF TYPE 3 RECORDS
1828	17A8	1828 17A8
1828	17A8	REM *****
1828	17A8	REM **** RECOR TYPE "4" - RECKEY PLUS OTHER CODES ****
1828	17A8	REM *****
1828	17A8	1828 17A8
1828	17A8	IF LEN(OTH.BUF\$)=0 THEN GOTO 224
1839	17A8	1839 17A8
1839	17A8	RPOINT=1
1840	17A8	220 RECOD4\$=MID\$(OTH.BUF\$,RPOINT,5)
1853	17AC	IF RECOD4\$="" THEN GOTO 222
1865	17AC	1865 17AC
1865	17AC	RECOU4\$=PGMID\$+"4"+RECKEY\$+RECOD4\$ 'TRANSACTION ID PLUS RECORD

Offset	Data	Source Line
187F	17AC	GOSUB 280
1884	17AC	PRINT #1,RECOU\$
188F	17AC	
188F	17AC	222 RPOINT = RPOINT + 5
1899	17AC	IF RPOINT < LEN(OTH.BUF\$) THEN GOTO 220
18AC	17AC	
18AC	17AC	
18AC	17AC	224 REM END OF TYPE 4 RECORDS
18AD	17AC	
18AD	17AC	REM *****
18AD	17AC	REM **** RECORD TYPE "5" - RECKEY PLUS GROUP I DATA ****
18AD	17AC	REM *****
18AD	17AC	
18AD	17AC	REM *****
18AD	17AC	REM **** RECORD TYPE "6" - RECKEY PLUS SPECIAL PROGRAMS ****
18AD	17AC	REM *****
18AD	17AC	
18AD	17AC	REM *****
18AD	17AC	REM **** RECORD TYPE "7" - RECKEY PLUS SECOND DX CODE ****
18AD	17AC	REM *****
18AD	17AC	
18AD	17AC	IF LEN(DX2.BUF\$)=0 THEN GOTO 268
18BE	17AC	
18BE	17AC	RPOINT=1
18C5	17AC	264 RECOD7\$=MID\$(DX2.BUF\$,RPOINT,5)
18D8	17B0	IF RECOD7\$="" THEN GOTO 266
18EA	17B0	
18EA	17B0	RECOU\$=PGMID\$+"7"+RECKEY\$+RECOD7\$ 'TRANSACTION ID 7
1904	17B0	GOSUB 280
1909	17B0	PRINT #1,RECOU\$
1914	17B0	
1914	17B0	266 RPOINT = RPOINT + 5
191E	17B0	IF RPOINT < LEN(DX2.BUF\$) THEN GOTO 264
1931	17B0	
1931	17B0	268 REM END OF TYPE 7 RECORDS
1932	17B0	
1932	17B0	REM *****
1932	17B0	REM **** RECORD TYPE "9" - RECKEY PLUS DISPOSITIONS ****
1732	17B0	REM *****
1932	17B0	IF LEN(DISPO\$)=0 THEN 278
193F	17B0	
193F	17B0	RPOINT=1
1946	17B0	274 RECOD9\$=MID\$(DISPO\$,RPOINT,2)
1959	17B4	IF RECOD9\$="" THEN 276
1967	17B4	
1967	17B4	RECOU\$=PGMID\$ + "9" + RECKEY\$ + RECOD9\$
1981	17B4	GOSUB 280
1986	17B4	PRINT #1,RECOU\$
1991	17B4	
1991	17B4	276 RPOINT=RPOINT+2
199A	17B4	IF RPOINT < LEN(DISPO\$) THEN 274
19A5	17B4	
19A6	17B4	278 REM END OF TYPE 9 RECORDS
19A7	17B4	

RACP900

OCC HEALTH FORM DESTING/DECODE PROGRAM

PAGE 13

07-06-87

15:02:00

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
19A7	17B4	GOTO 299
19AB	17B4	
19AB	17B4	REM *****
19AB	17B4	REM **** SUBROUTINE 270 - BUILD THE RECORD KEY ****
19AB	17B4	REM *****
19AB	17B4	270 RECKEY\$=""
19B4	17B4	
19B4	17B4	REM *** CLINIC ID (PREFIX + COD) ***
19B4	17B4	RECKEY\$= CL1.COD\$
19BD	17B4	
19BD	17B4	REM *** VISIT DATE ***
19BD	17B4	RECKEY\$=RECKEY\$+ VIDATES
19C9	17B4	
19C9	17B4	REM *** PRIMARY PROVIDER ***
19C9	17B4	RECKEY\$ = RECKEY\$ + PPROV.PFX\$ + PPROV.NUM\$
19DC	17B4	
19DC	17B4	REM *** PATIENT SSN ***
19DC	17B4	RECKEY\$ = RECKEY\$ + SSNS
19EB	17B4	
19EB	17B4	REM *** FAMILY MEMBER PREF ***
19EB	17B4	RECKEY\$ = RECKEY\$ + FMEHPS
19F4	17B4	
19F4	17B4	REM *** LITHO CODE ***
19F4	17B4	RECKEY\$ = RECKEY\$ + LITHOS
1A00	17B4	
1A00	17B4	REM *** FORM NUMBER ***
1A00	17B4	RECKEY\$ = RECKEY\$ + FRMNS
1A0C	17B4	
1A0C	17B4	RETURN
1A0F	17B4	
1A0F	17B4	REM *****
1A0F	17B4	REM **** SUBROUTINE 272 - BUILD THE DATA FOR TYPE 1 ****
1A0F	17B4	REM *****
1A0F	17B4	272 RECOD1\$=""
1A18	17B4	
1A18	17B4	REM *** VISIT COUNT ***
1A18	17B4	RECOD1\$ = RECOD1\$ + VCNT\$
1A24	17B4	
1A24	17B4	REM *** PRIMARY PROV TIME ***
1A24	17B4	RECOD1\$ = RECOD1\$ + PPROV.TIM\$
1A30	17B4	
1A30	17B4	REM *** SECONDARY PROVIDER ***
1A30	17B4	RECOD1\$ = RECOD1\$ + SPROV.PFX\$ + SPROV.NUM\$
1A43	17B4	
1A43	17B4	REM *** SECONDARY PROVIDER TIME ***
1A43	17B4	RECOD1\$ = RECOD1\$ + SPROV.TIM\$
1A4F	17B4	
1A4F	17B4	REM *** REASON FOR SECONDARY PROVIDER ***
1A4F	17B4	RECOD1\$ = RECOD1\$ + SPROV.REAS
1A5B	17B4	
1A5B	17B4	REM *** APPORINTMENT STATUS ***
1A5B	17B4	RECOD1\$ = RECOD1\$ + NOTSCS
1A67	17B4	
1A67	17B4	REM *** REFERRAL CODE ***

Offset Data Source Line

```

1A67 17B4      RECOD1$ = RECOD1$ + REF.PFX$ + REF.COD$
1A7A 17B4
1A7A 17B4      REM *** PLACE OF VISIT ***
1A7A 17B4      RECOD1$ = RECOD1$ + NOTCLS
1A86 17B4
1A86 17B4      REM *** JOB RELATED VISIT ***
1A86 17B4      RECOD1$ = RECOD1$ + JOBRIFS
1A92 17B4
1A92 17B4      REM *** MIL ONLY DUTY ***
1A92 17B4      RECOD1$ = RECOD1$ + MILDUT$
1A9E 17B4
1A9E 17B4      REM *** MIL ONLY QTRS ***
1A9E 17B4      RECOD1$ = RECOD1$ + MILQTRS
1AAA 17B4
1AAA 17B4      REM *** MIL ONLY PROFILE ***
1AAA 17B4      RECOD1$ = RECOD1$ + MILPRO$
1AB6 17B4
1AB6 17B4      REM *** NOT AVAILABLE ***
1AB6 17B4      RECOD1$ = RECOD1$ + NAVAIL$
1AC2 17B4
1AC2 17B4      REM *** ADMITTED ***
1AC2 17B4      RECOD1$ = RECOD1$ + ADMITS
1ACE 17B4
1ACE 17B4      REM *** ILLNESS ***
1ACE 17B4      RECOD1$ = RECOD1$ + ILLNS
1ADA 17B4
1ADA 17B4      REM *** INJURY ***
1ADA 17B4      RECOD1$ = RECOD1$ + INJUS
1AE6 17B4
1AE6 17B4      REM *** PURPOSE OF VISIT) ***
1AE6 17B4      RECOD1$ = RECOD1$ + PURPS
1AF2 17B4
1AF2 17B4      REM *** PRIM FOLLOW-UP/RULE OUT ***
1AF2 17B4      RECOD1$ = RECOD1$ + FU.ROS
1AFE 17B4
1AFE 17B4      REM *** PRIMARY DX CODE ***
1AFE 17B4      RECOD1$ = RECOD1$ + PRIMDX$
1B0A 17B4
1B0A 17B4      RETURN
1B0D 17B4
1B0D 17B4      REM *****
1B0D 17B4      REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
1B0D 17B4      REM *****
1B0D 17B4      280 PAD=MAXLENGTH - LEN(RECOUT$) 'FIND OUT HOW SHORT THE RECORD IS
1B1B 17B6      RECOD1$ = RECOD1$ + STRING$(PAD,PAD$) 'PAD THE RECORD WITH FILL CHAR
1B2E 17B6      RETURN
1B31 17B6
1B31 17B6      299 REM
1B32 17B6
1B32 17B6      REM -----END OF MODULE RACP900.MO2-----
1B32 17B6
1B32 17B6      998 REM CONTINUE
1B33 17B6
1B33 17B6      999 READTYPE = 2

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.06

```

183A 1786      IF LNLCOUNT > 48 THEN GOSUB 7100  'PRINTER HEADING
184A 1788      GOTO 10
184E 1788
184E 1788      REM END OF SCAN/DECODE/WRITE LOOP =====
184E 1788
184E 1788      1000 REM $INCLUDE: 'RACS1000.SUB'  INCLUDE THE VERIFY LOGON SUB
184F 1788      REM =====
184F 1788      REM * NAME: RACS1000          LOGON VERIFICATION SUBROUTINE      *
184F 1788      REM * Date: 28 Feb 84          PATIENT REGISTRATION PROGRAM      *
184F 1788      REM =====
184F 1788      REM          PATIENT OMR INPUT PROGRAM          *
184F 1788      REM          *
184F 1788      REM This program verifies user is logged on properly.  If there is no *
184F 1788      REM valid user logged on at the time of execution, this subroutine will*
184F 1788      REM chain to the logon program RACP05, otherwise a return is issued.  *
184F 1788      REM =====
184F 1788      REM          RESERVED LINE NUMBERS ARE 1001 THRU 1010
184F 1788      REM =====
184F 1788      1001 OPEN "I",1,"RACLOG.DAT"
1861 1788      IF EOF(1) THEN 1002                      'MAKE THEM LOG ON FIRST
186F 1788      INPUT #1,USERS(1),DTS,TMS,PIDS
1890 17C4      IF USERS(1) = "" THEN 1002                'MAKE THEM LOG ON FIRST
189E 17C4      IF USERS(1) = "*****" THEN 1002          'MAKE THEM LOG ON FIRST
18AC 17C4      CLOSE 1
18B3 17C4      SCREEN 0,1,0,0
18C9 17C4      COLOR FORE,BACK,BORD
18DF 17C4      CLS
18E3 17C4      RETURN
18E6 17C4
18E6 17C4      1002 CLOSE
18EA 17C4      CHAIN "RACP05"
18F1 17C4      '=====END OF LOGON VERIFY SUBROUTINE 1000=====
18F1 17C4
18F1 17C4      2000 REM $INCLUDE: 'RACS2000.SUB'  INCLUDE THE RPLY/DELAY SUB
18F2 17C4      REM =====
18F2 17C4      REM ****      AMBULATORY CARE DATA BASE          13 APR 85      ****
18F2 17C4      REM ****      SKIP COLE                          ****
18F2 17C4      REM ****      SUBROUTINE NAME      :      RACS2000.SUB      ****
18F2 17C4      REM ****      SCANNER PROGRAM #    :      ALL                ****
18F2 17C4      REM ****      FUNCTION              :      THIS SUBROUTINE MODULE ****
18F2 17C4      REM ****      SERVERS AS A WAIT AND REPLY ****
18F2 17C4      REM ****      ENTRY MODULE          ****
18F2 17C4      REM ****      INPUT                  :      SINGLE KEYBOARD ENTRY ****
18F2 17C4      REM ****      OUTPUT                 :      KEYBOARD ENTRY - UPPER CASE ****
18F2 17C4      REM ****      RESERVED LINE
18F2 17C4      REM ****      NUMBERS              :      2001-2010          ****
18F2 17C4      REM =====
18F2 17C4      2001 REM REPLY FUNCTION
18F3 17C4      2002 REPLY$=INKEY$ : IF REPLY$="" THEN 2002
1C07 17C8      REPLY=ASC(REPLY$)
1C11 17CA      IF REPLY > 90 THEN REPLY$=CHR$(REPLY XOR 32)  'CONVERT TO CAPS
1C2C 17CA      IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$="?"

```

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line	
1C58	17CA	RETURN	
1C5B	17CA		
1C5B	17CA	5000 REM \$INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB	
1C5C	17CA	REM *****	
1C5C	17CA	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****	
1C5C	17CA	REM **** SKIP COLE ****	
1C5C	17CA	REM **** SUBROUTINE NAME : RXXS5000.SUB ****	
1C5C	17CA	REM **** SCANNER PROGRAM # : ALL ****	
1C5C	17CA	REM **** FUNCTION : THIS SUBROUTINE MODULE ****	
1C5C	17CA	REM **** PERFORMS A DATE EDIT ****	
1C5C	17CA	REM ****	
1C5C	17CA	REM **** INPUT : DATE TO BE CHECKED MUST BE ****	
1C5C	17CA	REM **** IN THE VARIABLE NAMED ****	
1C5C	17CA	REM **** 'CK.5000\$' ****	
1C5C	17CA	REM **** IN THE FORMAT "YYMMDD" ****	
1C5C	17CA	REM ****	
1C5C	17CA	REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****	
1C5C	17CA	REM **** VARIABLE. IF THIS VARIABLE ****	
1C5C	17CA	REM **** CONTAINS ANY NUMBER OTHER ****	
1C5C	17CA	REM **** THAN 0, AN ERROR WAS FOUND ****	
1C5C	17CA	REM **** IN THE DATE. ****	
1C5C	17CA	REM ****	
1C5C	17CA	REM **** RESERVED LINE ****	
1C5C	17CA	REM **** NUMBERS : 5001-5009 ****	
1C5C	17CA	REM *****	
1C5C	17CA	RT.5000 = 0	
1C63	17CA	CKYEAR = VAL(LEFT\$(CK.5000\$,2)) 'YEAR NUMERIC VALUE	
1C76	17CC	CKMONTH = VAL(MID\$(CK.5000\$,3,2)) 'MONTH NUMERIC VALUE	
1C8C	17CE	CKDAY = VAL(RIGHT\$(CK.5000\$,2)) 'DAY NUMERIC VALUE	
1C9F	17D0		
1C9F	17D0	IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009	
1CB5	17D0	IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009	
1CCB	17D0	IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009	
1CE1	17D0	IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009	
1CF7	17D0		
1CF7	17D0	REM LEAP YEAR CHECK	
1CF7	17D0	MOLENGTH(2) = 28	
1CFE	17D0	IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY	
1D0D	17D0	IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR	
1D22	17D0	MOLENGTH(2) = 29	
1D29	17D0		
1D29	17D0	5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009	
1D48	17D0		
1D48	17D0	5009 RETURN	
1D4B	17D0		
1D4B	17D0	REM -----END OF SUBROUTINE 5000 -----	
1D4B	17D0		
1D4B	17D0	5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTP UCA VALIDATE SUB	
1D4C	17D0	5600 REM INCLUDE: 'RACS5600.SUB' INCLUDE THE INP UCA VALIDATE SUB	
1D4D	17D0	5700 REM \$INCLUDE: 'RACS5700.SUB' INCLUDE MAP ONES TO POSITION NO.	
1D4E	17D0	REM *****	
1D4E	17D0	REM **** AMBULATORY CARE DATA BASE 29 JUL 85 ****	
1D4E	17D0	REM **** D R BOLLING ****	

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

104E 17D0 REM **** SUBROUTINE NAME      :  RXXS5700.SUB      ****
104E 17D0 REM **** SCANNER PROGRAM #    :  ALL                ****
104E 17D0 REM **** FUNCTION              :  THIS SUBROUTINE MODULE ****
104E 17D0 REM ****                      :  CONVERTS A BINARY ARRAY INTO ****
104E 17D0 REM ****                      :  TWO CHAR CODES.      ****
104E 17D0 REM ****                      :                      ****
104E 17D0 REM **** INPUT                :  INP.STO$ AS STRING ****
104E 17D0 REM ****                      :                      ****
104E 17D0 REM ****                      :                      ****
104E 17D0 REM **** OUTPUT              :  BUF.STO$ AS STRING. ****
104E 17D0 REM ****                      :                      ****
104E 17D0 REM ****                      :                      ****
104E 17D0 REM **** RESERVED LINE      :                      ****
104E 17D0 REM **** NUMBERS              :  5710-5730          ****
104E 17D0 REM ****                      :                      ****
104E 17D0 REM ****                      :                      ****
104E 17D0
104E 17D0 BUF.STO$=""
1057 17D0 N.STO=1
105E 17D0 5710 X.STO=INSTR(N.STO,INP.STO$,"1")
1070 17D0 IF X.STO=0 THEN GOTO 5720 'THATS ALL
107F 17D0 N.STO = X.STO + 1 'NEXT STARTING POINT
1087 17D0 X.STO = X.STO + 100 'PAD WITH LEADING ZERO
1091 17D0 BUF.STO$ = BUF.STO$ + RIGHT$(STR$(X.STO),2)
10A7 17D0 IF N.STO <= LEN(INP.STO$) THEN GOTO 5710
10BA 17D0
10BA 17D0 5720 RETURN
10BD 17D0
10BD 17D0 REM ----- END OF SUBROUTINE 5700 -----
10BD 17D0 5800 REM $INCLUDE: 'RACS5800.SUB' INCLUDE FOUR COL DX CONVERTER
10BE 17D0 REM *****
10BE 17D0 REM *** AMBULATORY CARE DATA BASE 30 JUL 85 ***
10BE 17D0 REM *** D R BOLLING ***
10BE 17D0 REM *** SUBROUTINE NAME      :  RXXS5800.SUB      ***
10BE 17D0 REM *** SCANNER PROGRAM    :  AS APPROPRIATE      ***
10BE 17D0 REM *** FUNCTION          :  THIS SUBROUTINE MODULE READS ***
10BE 17D0 REM ***                      :  FOUR COLUMNS IN DX AREA AND ***
10BE 17D0 REM ***                      :  CONVERTS TO A POSITION IN A ***
10BE 17D0 REM ***                      :  TABLE. AN ERROR CODE IS ***
10BE 17D0 REM ***                      :  RETURNED IF MULTIPLE CODES ARE ***
10BE 17D0 REM ***                      :  PRESENT. ***
10BE 17D0 REM ***                      :                      ***
10BE 17D0 REM *** INPUT              :  X.POS - STARTING POSITION IN ***
10BE 17D0 REM ***                      :  STRING ***
10BE 17D0 REM ***                      :                      ***
10BE 17D0 REM *** OUTPUT            :  X.FIN - TABLE POSITION OF DX ***
10BE 17D0 REM ***                      :  RT.5800 = 1 ON ERROR ***
10BE 17D0 REM ***                      :                      ***
10BE 17D0 REM *** RESERVED LINE NUMBERS :  5801 - 5899      ***
10BE 17D0 REM *****
10BE 17D0 RT.5800=0 'INITIALIZE ERROR INDICATOR
10C5 17D0 X.FIN=0 'INITIALIZE TABLE OFFSET RESULT
10CC 17D0
10CC 17D0 REM ** COLUMN 1 **
10CC 17D0 X=0 'STARTING POINTER

```

Offset	Data	Source Line
1003	1700	X.SIZ=C1SIZ
100A	1702	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
10F1	1702	IF X.STO=0 THEN GOTO 5802
1E00	1702	X.FIN=X.STO
1E07	1702	
1E07	1702	5802 REM ** COLUMN 2 **
1E08	1702	IF C2SIZ=0 THEN GOTO 5804
1E17	1702	X=X+X.SIZ
1E22	1702	X.POS=X.POS+2
1E2B	1702	X.SIZ=C2SIZ
1E32	1702	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
1E49	1702	IF X.STO=0 THEN GOTO 5804
1E58	1702	IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
1E67	1702	X.FIN=X.STO + X
1E72	1702	
1E72	1702	5804 REM ** COLUMN 3 **
1E73	1702	IF C3SIZ=0 THEN GOTO 5806
1E82	1702	X=X+X.SIZ
1E8D	1702	X.POS=X.POS+2
1E96	1702	X.SIZ=C3SIZ
1E9D	1702	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
1EB4	1702	IF X.STO=0 THEN GOTO 5806
1EC3	1702	IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
1ED2	1702	X.FIN=X.STO + X
1EDD	1702	
1EDD	1702	5806 REM ** COLUMN 4 **
1EDE	1702	IF C4SIZ=0 THEN GOTO 5808
1EED	1702	X=X+X.SIZ
1EF8	1702	X.POS=X.POS+2
1F01	1702	X.SIZ=C4SIZ
1F08	1702	X.STO=VAL(MID\$(TEXT\$,X.POS,2))
1F1F	1702	IF X.STO=0 THEN GOTO 5808
1F2E	1702	IF X.FIN<>0 THEN GOTO 5890 'ERROR - MULTIPLE CODE
1F3D	1702	X.FIN=X.STO + X
1F48	1702	
1F48	1702	5808 REM
1F49	1702	GOTO 5899
1F4D	1702	
1F4D	1702	5890 RT.5800=1 'ERROR - MULTIPLE CODES
1F54	1702	
1F54	1702	5899 RETURN
1F57	1702	
1F57	1702	REM -----END OF SUBROUTINE RXXS5800.SUB-----
1F57	1702	
1F57	1702	6000 REM \$INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE SUB
1F58	1702	REM *****
1F58	1702	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1F58	1702	REM **** SKIP COLE ****
1F53	1702	REM **** SUBROUTINE NAME : RXXS6000.SUB ****
1F58	1702	REM **** SCANNER PROGRAM # : ALL ****
1F58	1702	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1F58	1702	REM **** PERFORMS INSTRING SEARCH ****
1F58	1702	REM ****

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

1F58 17D2 REM **** INPUT : STRING TO BE SEARCHED MUST ****
1F58 17D2 REM **** BE IN THE VARIABLE NAMED : ****
1F58 17D2 REM **** 'X$' ****
1F58 17D2 REM ****
1F58 17D2 REM **** OUTPUT : 'TOT' = TOTAL NUMBER OF ****
1F58 17D2 REM **** HITS IN THE DESTING ****
1F58 17D2 REM **** 'HOLD$( )' IS THE ARRAY ****
1F58 17D2 REM **** CONTAINING THE NUMERIC ****
1F58 17D2 REM **** VALUE OF THE HIT POSITIONS ****
1F58 17D2 REM ****
1F58 17D2 REM *****
1F58 17D2 REM **** RESERVED LINE ****
1F58 17D2 REM **** NUMBERS : 6001-6009 ****
1F58 17D2 REM *****
1F58 17D2 6001 PTR = INSTR(X$, "1")
1F66 17D4 TOT = 0
1F6D 17D6 WHILE PTR > 0
1F78 17D6 TOT=TOT+1
1F80 17D6 HOLD$(TOT) = RIGHT$(STR$(PTR),2)
1FA2 17D6 PTR=PTR+1
1FAA 17D6 PTR = INSTR(PTR,X$, "1")
1FBC 17D6 WEND
1FC0 17D6 RETURN
1FC3 17D6
1FC3 17D6 REM -----END OF SUBROUTINE RXXS6000.SUB-----
1FC3 17D6
1FC3 17D6
1FC3 17D6 7000 REM $INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
1FC4 17D6 REM *****
1FC4 17D6 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1FC4 17D6 REM **** SKIP COLE ****
1FC4 17D6 REM **** SUBROUTINE NAME : RACS7000.SUB ****
1FC4 17D6 REM **** SCANNER PROGRAM # : ALL ****
1FC4 17D6 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1FC4 17D6 REM **** PRINTS THE STANDARD SCREEN ****
1FC4 17D6 REM **** HEADING. ****
1FC4 17D6 REM **** INPUT : COMMON VARIABLE USERS$(2) ****
1FC4 17D6 REM **** SYSTEM DATE ****
1FC4 17D6 REM ****
1FC4 17D6 REM **** OUTPUT : SCREEN HEADING ****
1FC4 17D6 REM ****
1FC4 17D6 REM **** RESERVED LINE ****
1FC4 17D6 REM **** NUMBERS : 7001-7010 ****
1FC4 17D6 REM *****
1FC4 17D6
1FC4 17D6 7001 LOCATE 1,1
1FCE 17D6 PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
1FD6 17D6 LOCATE 1,65
1FE3 17D6 PRINT DATES;
1FEB 17D6 LOCATE 2,1
1FF8 17D6 PRINT "USER : ";USERS$(1)
2005 17D6 RETURN
2008 17D6
2008 17D6 7100 REM $INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

2009 17D6 REM *****
2009 17D6 REM ****    AMBULATORY CARE DATA BASE          13 APR 85    ****
2009 17D6 REM ****                                SKIP COLE      ****
2009 17D6 REM ****    SUBROUTINE NAME      :    RXXS7100.SUB          ****
2009 17D6 REM ****    SCANNER PROGRAM #    :    ALL                  ****
2009 17D6 REM ****    FUNCTION              :    THIS SUBROUTINE MODULE ****
2009 17D6 REM ****                                PRINTS THE STANDARD HEADING ****
2009 17D6 REM ****                                ON THE PRINTER.      ****
2009 17D6 REM ****    INPUT                  :    DATE,PAGE,PGMIDS,PGMTITLS ****
2009 17D6 REM ****                                ****
2009 17D6 REM ****    OUTPUT                  :    PRINTER HEADING, LN.COUNT ****
2009 17D6 REM ****                                ****
2009 17D6 REM ****    RESERVED LINE
2009 17D6 REM ****            NUMBERS      :    7101-7110          ****
2009 17D6 REM *****
2009 17D6 7101 IF PAGE > 0 THEN LPRINT CHR$(12);
201F 17D6      LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITLS;
202C 17DA      LPRINT TAB(70);DATES
203F 17DA      PAGE=PAGE+1
2047 17DA      LPRINT "PROGRAM ";PGMIDS;TAB(70);"PAGE";
2064 17DA      LPRINT USING "####";PAGE
2070 17DA      LPRINT
2078 17DA      LN.COUNT=3
207F 17DA      RETURN
2382 17DA
2082 17DA 8000 REM $INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
2083 17DA REM *****
2083 17DA REM ****    AMBULATORY CARE DATA BASE          13 APR 85    ***
2083 17DA REM ****                                SKIP COLE      ***
2083 17DA REM ****    SUBROUTINE NAME      :    RXXS8000.SUB          ***
2083 17DA REM ****    SCANNER PROGRAM #    :    ALL                  ***
2083 17DA REM ****    FUNCTION              :    THIS SUBROUTINE MODULE ***
2083 17DA REM ****                                IS A GROUPING THAT PERFORMS ***
2083 17DA REM ****                                VARIOUS DECODING FUNCTIONS ***
2083 17DA REM ****                                ON THE SCANNER DATA ***
2083 17DA REM ****                                ****
2083 17DA REM ****    8001    - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
2083 17DA REM ****    8050    - CHECK FOR END OF JOB                ***
2083 17DA REM ****    8100    - PRINT THE HEADER DATA ON THE SCREEN ***
2083 17DA REM ****    8200    - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
2083 17DA REM ****                                (RETURNED IN TEXT$ STRING VARIABLE) ***
2083 17DA REM ****                                ****
2083 17DA REM ****    INPUT                  :    SHEET RECORD, RECORD LENGTH ***
2083 17DA REM ****                                ****
2083 17DA REM ****    OUTPUT                  :    'TEXT$' TRING VARIABLE ***
2083 17DA REM ****                                ****
2083 17DA REM ****    RESERVED LINE
2083 17DA REM ****            NUMBERS      :    8001-8500          ***
2083 17DA REM *****
2083 17DA 'DECODE THE HEADER ONLY
2083 17DA 8001    POINTER = 0
2083 17DA RECORDPTR = VARPTR(SHEETREC(0))

```

Offset	Data	Source Line
2091	17DE	FOR J8000 = 1 TO 21
2098	17DE	8002 TEXTS= TEXTS+CHR\$(PEEK(RECORDPTR + POINTER))
20B6	17DE	POINTER=POINTER+1
20BE	17DE	NEXT J8000
20CD	17E0	PROGRAMS= LEFT\$(TEXT\$,3)
20DC	17E4	BATCHS= MID\$(TEXT\$,4,3)
20EE	17E8	SERIALS= MID\$(TEXT\$,7,4)
2100	17EC	RUNIDS= MID\$(TEXT\$,11,1)
2112	17F0	FORMS= MID\$(TEXT\$,12,2)
2124	17F4	POCKETS= MID\$(TEXT\$,14,1)
2136	17F8	SCANERR1S=MID\$(TEXT\$,16,2)
2148	17FC	SCANERR2S=MID\$(TEXT\$,18,2)
215A	1800	SCANERR3S=MID\$(TEXT\$,20,2)
216C	1804	GOTO 8500
2170	1804	
2170	1804	8050 REM CHECK FOR END OF JOB/END OF BATCH
2171	1804	IF PROGRAMS = PGMID\$ THEN GOTO 8500
2183	1804	LPRINT STRINGS(80,"")
2191	1804	LPRINT
2199	1804	LPRINT "RECORDS PROCESSED ... ";SERIALS
21A6	1804	LPRINT "STARTED AT ";BTIMES
21B3	1804	LPRINT "ENDED AT ";TIMES
21C0	1804	LPRINT CHR\$(12)
21C8	1804	GOTO 30000
21CF	1804	
21CF	1804	8070 REM CHECK FOR SCANNER ERRORS
21D0	1804	IF POCKETS = " " GOTO 8500
21E2	1804	LPRINT LITHOS;
21EA	1804	LPRINT " ... SCANNER ERRORS : ";
21F2	1804	LPRINT SCANERR1S;" / ";
21FF	1804	LPRINT SCANERR2S;" / ";
220C	1804	LPRINT SCANERR3S
2214	1804	LN=LN+1
221C	1806	GOTO 999
2220	1806	
2220	1806	8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
2221	1806	LOCATE 5,1:PRINT "PROGRAM ";PROGRAMS;
2238	1806	
2238	1806	PRINT " BATCH ";BATCHS;
2248	1806	PRINT " RUN ";RUNIDS;
2255	1806	PRINT " FORM ";FORMS;
2262	1806	PRINT " POCKET ";POCKETS
226F	1806	GOTO 8500
2273	1806	
2273	1806	8200 REM DECODE THE RESPONSE POSITIONS
2274	1806	POINTER = 21
2278	1806	RECORDPTR = VARPTR(SHEETREC(0))
2282	1806	FOR J8000 = 22 TO RECORDLENGTH
228F	1808	8202 TEXTS = TEXTS+CHR\$(PEEK(RECORDPTR + POINTER))
22AD	1808	POINTER=POINTER+1
22B5	1808	NEXT J8000
22C6	1808	
22C6	1808	8500 RETURN
22C9	1808	

```

Offset  Data  Source Line
22C9 1808      REM ----- END OF RXXS8000.SUB -----
22C9 1808
22C9 1808 9000 REM $INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
22CA 1808 REM *****
22CA 1808 REM ****    AMBULATORY CARE DATA BASE            13 APR 85    ****
22CA 1808 REM ****                                SKIP COLE        ****
22CA 1808 REM ****    PROGRAM NAME      :    RACS9000.SUB          ****
22CA 1808 REM ****    SCANNER PROGRAM #  :    ALL                  ****
22CA 1808 REM ****    FUNCTION          :    THIS SUBROUTINE MODULE  ****
22CA 1808 REM ****                                CONTROLS THE SCANNER I/O  ****
22CA 1808 REM ****                                ****
22CA 1808 REM ****    INPUT/OUTPUT      :    REFER TO THE ASYNCHRONOUS  ****
22CA 1808 REM ****                                COMMUNICATIONS MANUAL AND THE  ****
22CA 1808 REM ****                                PRE-RELEASED SOFTWARE GUIDE  ****
22CA 1808 REM ****                                ****
22CA 1808 REM *****
22CA 1808 REM ****    RESERVED LINE                                ****
22CA 1808 REM ****                                ****
22CA 1808 REM ****                                NUMBERS      :    9001-9100    ****
22CA 1808 REM *****
22CA 1808
22CA 1808
22CA 1808 REM *****
22CA 1808 REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER          ****
22CA 1808 REM **** ARGUMENTS: PRESET ... SEE BELOW                      ****
22CA 1808 REM *****
22CA 1808 9001 REM
22CB 1808      PROTOCOL(0) = 9600          'BAUD RATE
22D2 1808      PROTOCOL(1) = 78           'PARITY (SEE PAGE 4-8 OF MANUAL)
22D9 1808      PROTOCOL(2) = 8           'DATA BITS
22E0 1808      PROTOCOL(3) = 1           'STOP BITS
22E7 1808      PROTOCOL(4) = 2           'RS-232 PORT
22EE 1808      PROTOCOL(5) = 0           'WRITE TIME-OUT
22F5 1808      PROTOCOL(6) = 0           'READ TIME-OUT
22FC 1808
22FC 1808      ERRSTAT$ = SPACES(60)
2308 1808      ARGPTR = VARPTR(PROTOCOL(0))
230F 180A      CALL SETUP (ARGPTR,ERRSTAT$)
2320 180A      ERRMSG$=""
2329 180A      IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="SETUP ERROR "+ERRSTAT$
2345 180A      GOTO 9100
2349 180A
2349 180A REM *****
2349 180A REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER          ****
2349 180A REM **** ARGUMENTS: CNTRLOPT                                ****
2349 180A REM **** CNTRLOPT = 1 = START SCANNER (S1)                  ****
2349 180A REM **** CNTRLOPT = 2 = STOP SCANNER (S0)                  ****
2349 180A REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
2349 180A REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2)        ****
2349 180A REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31"      ****
2349 180A REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32"    ****
2349 180A REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
2349 180A REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC)              ****
2349 180A REM *****
2349 180A 9010 REM

```

Offset	Data	Source Line
234A	180A	ERRSTAT\$ = SPACES(60)
2356	180A	CALL CNTRL (CNTRLOPY,ERRSTAT\$)
2367	180A	ERRMSG\$=""
2370	180A	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
238C	180A	GOTO 9100
2390	180A	
2390	180A	REM *****
2390	180A	REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
2390	180A	REM ****
2390	180A	REM **** ARGUMENTS: READTYPE ****
2390	180A	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
2390	180A	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
2390	180A	REM ****
2390	180A	REM **** ARGUMENTS: RECORDLENGTH ****
2390	180A	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
2390	180A	REM **** TRANSMITTED ****
2390	180A	REM *****
2390	180A	9020 REM
2391	180A	ERRSTAT\$ = SPACES(60)
2390	180A	RECORDPTR = VARPTR(SHEETREC(0))
23A4	190A	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
23B0	180A	ERRMSG\$=""
23C6	180A	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
23E7	180A	GOTO 9100
23EB	180A	
23EB	180A	REM *****
23EB	180A	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
23EB	180A	REM ****
23EB	180A	REM **** ARGUMENTS: PRINTPOS ****
23EB	180A	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
23EB	180A	REM **** VALUES = 0 THRU 90 ****
23EB	180A	REM ****
23EB	180A	REM **** ARGUMENTS: PSTRINGS ****
23EB	180A	REM **** TEXT TO BE PRINTED ON THE FORM ****
23EB	180A	REM ****
23EB	180A	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
23EB	180A	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
23EB	180A	REM *****
23EB	180A	9030 REM
23EC	180A	ERRSTAT\$ = SPACES(60)
23F8	180A	RECORDPTR = VARPTR(SHEETREC(0))
23FF	180A	CALL TPRINT(PRINTPOS,PSTRINGS\$,ERRSTAT\$)
2414	1810	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
2430	1810	GOTO 9100
2434	1810	
2434	1810	9100 RETURN
2437	1810	REM -----END OF SUBROUTINE RACS9000.SUB -----
2437	1810	
2437	1810	REM END OF SUBROUTINES =====
2437	1810	
2437	1810	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
2438	1810	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
2440	1810	LPRINT "ERASING FILE ";DATFIL\$
2440	1810	BEEP

RACP900
OCC HEALTH FORM DESTURING/DECODE PROGRAM

PAGE 29

07-06-87

15:02:00

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

2451	1810	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
245D	1810	CLOSE
2461	1810	OPEN DATFIL\$ FOR OUTPUT AS #1
2473	1810	PRINT #1,STRING\$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
2485	1810	CLOSE
2489	1810	
2489	1810	30000 REM
248A	1810	CLOSE
248E	1810	CHAIN "RACP10"
2495	1810	
2498	1810	

22151 Bytes Available

14906 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

PAGE 1

04-28-87

09:33:34

IBM Personal Computer BASIC Compiler /1.00

Offset Data Source Line

001A	0002	REM \$LINESIZE: 132
001A	0002	REM \$PAGESIZE: 66
001A	0002	REM \$TITLE: 'RACP910 '
001A	0002	REM \$SUBTITLE: 'OT REPEAT VISIT FORM'
001A	0002	REM \$PAGE

RACP910

PAGE 2

OT REPEAT VISIT FORM

04-28-87

09:33:34

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```

001A 0002 REM +-----+
001A 0002 REM | NAME: RACP910          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | DATE: 12 MAR 87      OT (DHBA) REPEAT VISIT      |
001A 0002 REM | D R BOLLING          SHORT FORM                  |
001A 0002 REM |
001A 0002 REM | INCLUDES PREFIX TO LITHO FOR EACH PATIENT ON FORM |
001A 0002 REM +-----+
001A 0002 REM          OT REPEAT FORM INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the SHORT form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RXXP05, otherwise,
001A 0002 REM THE PROGRAM CHAINS TO PROGRAM RACP10 ON EXIT.
001A 0002
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD          DIMENSION DEFINITIONS      *
001A 0002 '* Date: 28 Feb 84          Written by: Floyd Cole      *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' effect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002
001A 0002 DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 DIM ED.MSG$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002 DIM CLINIC2.PFX$(6) '(PREFIX -A B C D F S- FOR CLINIC #2)
001A 0002 DIM PROCED$(26) '(PROCEDURE TABLE FOR SHORT FORM)
001A 0002 DIM PROVIDER.TIMES$(8) '(PROVIDER TIME TABLE)

```

Offset	Data	Source Line
001A	0002	DIM REFCODS(18) '(REFERRAL CODE TABLE)
001A	0002	DIM VDATS(4) '
001A	0002	DIM PR1.CODS(4) '
001A	0002	DIM PR1.TIMS(4) ' ARRAYS TO HOLD MULTIPLE VISITS
001A	0002	DIM PR2.CODS(4) '
001A	0002	DIM PR2.TIMS(4) '
001A	0002	DIM SPROV.REAS(4) '
001A	0002	DIM VIS.CNTS(4) '
001A	0002	DIM TOT.PROC1(4) '
001A	0002	DIM TOT.PROC2(4) '
001A	0002	DIM GROUP1S(6,27) '(PATIENT/PROCEDURE GROUP PROV 1)
001A	0002	DIM GROUP2S(6,26) '(PATIENT/PROCEDURE GROUP PROV 2)
001A	0002	DIM HOLDS(26) '(HOLD AREA FOR SUBROUTINE 6000)
001A	0002	
001A	0002	REM \$INCLUDE: 'RACCMN.MOD' REM INCLUDE THE COMMON AREA DEFINITION
001A	0002	*****
001A	0002	* NAME: RACCMN.MOD COMMON AREA DEFINITION *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A	0002	' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	' *****START OF COMMON DEFINITIONS*****
001A	0002	
001A	0002	COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELLS 'BASIC SCREEN COLORS
001A	0002	COMMON HEADERS '21 CHARACTER SCANNER HEADER INFO
001A	0002	COMMON TEXTS ' AINING CHARACTERS FROM SCANNER
001A	0002	COMMON PGMIDS 'PROGRAM OR FORM ID
001A	0002	COMMON MOLENGTH() 'DAYS IN THE MONTH
001A	0002	COMMON USERS()
001A	0002	' *****END OF COMMON DEFINITION*****
001A	0002	
001A	0002	
001A	0002	REM \$INCLUDE: 'RACDEF.MOD' REM INCLUDE THE DEFAULT DEFINITIONS
001A	0002	*****
001A	0002	* NAME: RACP01.DEF DEFAULT DEFINITIONS *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' Variables used in common that have a default value on start*up
001A	0002	' will be held in this file. It is an included file so it cannot
001A	0002	' be run in a stand*alone mode. In normal operation, this file
001A	0002	' should be 'included' in the main program only (RACP10.BAS).
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	' *****START OF DEFAULT DEFINITION*****
001A	0002	
001A	0002	FORE = 15 'FOREGROUND COLOR = INTENSE WHITE

PAGE 4
04-28-87
09:33:34

IBM Personal Computer BASIC Compiler V1.00

```

0050      1654      BACK = 1          'Background Color = Light Blue
0057      1654      BORD = 4          'BORDER              = RED
005E      1656      HIDE = 4          'ALTERNATE COLOR = RED
0065      1656      EFOR= 14         'ERROR FOREGROUND DISPLAY
006C      1656      EBAC= 0          'ERROR BACKGROUND DISPLAY
0073      1656      BELLS = CHR$(7)   'Sound the bell
007F      1656
007F      1656      MOLENGTH(1) = 31    'JAN
0086      1656      MOLENGTH(2) = 28    'FEB ---MODIFIED IN SUBROUTINE RACS5000.SUB
008D      1656      MOLENGTH(3) = 31    'MAR
0094      1656      MOLENGTH(4) = 30    'APR
0098      1656      MOLENGTH(5) = 31    'MAY
00A2      1656      MOLENGTH(6) = 30    'JUN
00A9      1656      MOLENGTH(7) = 31    'JUL
00B0      1656      MOLENGTH(8) = 31    'AUG
00B7      1656      MOLENGTH(9) = 30    'SEP
00BE      1656      MOLENGTH(10) = 31   'OCT
00C5      1656      MOLENGTH(11) = 30   'NOV
00CC      1656      MOLENGTH(12) = 31   'DEC
00D3      1656
00D3      1656      DATEERR$(0) = " "
00DC      1656      DATEERR$(1) = "INVALID MONTH"
00E5      1656      DATEERR$(2) = "INVALID DAY "
00EE      1656      DATEERR$(3) = "DAY TOO LARGE FOR MONTH CODED"
00F7      1656
00F7      1656      MAXLENGTH = 80       'MAXIMUM LENGTH OF OUTPUT RECORD
00FE      1658      PAD$ = ". "        'PAD CHARACTER FOR SHORT RECORDS
0107      165C
0107      165C      ' *****END OF DEFAULT DEFINITION*****
0107      165C
0107      165C      KEY OFF
0100      165C
0100      165C      REM *****
0100      165C      REM THE FOLLOWING VARIABLES ARE UNIQUE TO EACH PROGRAM AND MUST
0100      165C      REM BE CHANGED.
0100      165C      REM *****
0100      165C      PGMTITL$ = "OT (DHBA) REPEAT FORM"
0116      1660
0116      1660      PGMDID$ = "910"        'VALUE RECEIVED FROM THE SCANNER
011F      1660      'IN HEADER VARIABLE 'PROGRAM$'
011F      1660
011F      1660      DATFIL$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
0128      1664
0128      1664      REM LENGTH OF STRING RECEIVED FROM THE OMR....
0128      1664      HEADER = 21
012F      1666      RESPONSE= 331
0136      1668      RECORDLENGTH = HEADER + RESPONSE
0141      166A
0141      166A      N.PROC = 26     ' NUMBER OF PROCEDURES FOR THIS FORM
0148      166C
0148      166C      REM *****
0148      166C
0148      166C      BTIME$=TIME$           'SCAN START TIME
0148      166C

```

Offset Data Source Line

```

0151 1670
0151 1670
0151 1670 REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0151 1670 REM CLINIC #2
0151 1670 CLINIC2.PFX$(0)=" "
015A 1670 CLINIC2.PFX$(1)="A"
0163 1670 CLINIC2.PFX$(2)="B"
016C 1670 CLINIC2.PFX$(3)="C"
0175 1670 CLINIC2.PFX$(4)="D"
017E 1670 CLINIC2.PFX$(5)="F"
0187 1670 CLINIC2.PFX$(6)="S"
0190 1670
0190 1670 REM *** PREASSIGNED REFERRAL CODES ***
0190 1670 REFCOD$(1)="AAAA" : REFCOD$(07)="AFYC" : REFCOD$(13)="BGYA"
01AB 1670 REFCOD$(2)="AAAF" : REFCOD$(08)="BAAA" : REFCOD$(14)="BHAE"
01C6 1670 REFCOD$(3)="ABAA" : REFCOD$(09)="BBAA" : REFCOD$(15)="BHAF"
01E1 1670 REFCOD$(4)="ABDA" : REFCOD$(10)="BBCA" : REFCOD$(16)="BHAG"
01FC 1670 REFCOD$(5)="AEAA" : REFCOD$(11)="BDAA" : REFCOD$(17)="BIYA"
0217 1670 REFCOD$(6)="AFYA" : REFCOD$(12)="BEAA" : REFCOD$(18)="DHDA"
0232 1670
0232 1670 REM *** ENCOUNTER FORM PROCEDURE TABLE ***
0232 1670 PROCED$(0)=" " : PROCED$(10) = "04224" : PROCED$(20)= "04125"
0240 1670 PROCED$(1)="04215" : PROCED$(11) = "04184" : PROCED$(21)= "04227"
0268 1670 PROCED$(2)="04214" : PROCED$(12) = "04104" : PROCED$(22)= "04085"
0283 1670 PROCED$(3)="04213" : PROCED$(13) = "04129" : PROCED$(23)= "04087"
029E 1670 PROCED$(4)="04136" : PROCED$(14) = "04094" : PROCED$(24)= "04194"
02B9 1670 PROCED$(5)="04192" : PROCED$(15) = "04208" : PROCED$(25)= "04126"
02D4 1670 PROCED$(6)="04092" : PROCED$(16) = "04207" : PROCED$(26)= "04127"
02EF 1670 PROCED$(7)="04128" : PROCED$(17) = "04186"
0301 1670 PROCED$(8)="04229" : PROCED$(18) = "04185"
0313 1670 PROCED$(9)="04196" : PROCED$(19) = "04095"
0325 1670
0325 1670 REM *** PROVIDER TIME TABLE ***
0325 1670 PROVIDER.TIME$(00)="000" ' NO TIME
032E 1670 PROVIDER.TIME$(01)="005" ' 5 MINUTES
0337 1670 PROVIDER.TIME$(02)="010" '10 MINUTES
0340 1670 PROVIDER.TIME$(03)="015" '15 MINUTES
0349 1670 PROVIDER.TIME$(04)="020" '20 MINUTES
0352 1670 PROVIDER.TIME$(05)="030" '30 MINUTES
035B 1670 PROVIDER.TIME$(06)="045" '45 MINUTES
0364 1670 PROVIDER.TIME$(07)="060" '60 MINUTES
036D 1670 PROVIDER.TIME$(08)="999" ' > 1 HOUR
0376 1670
0376 1670 REM INCLUDE: 'UCABAMC.OPT' INCLUDE OUTPATIENT UCA TABLE
0376 1670
0376 1670 REM $PAGE

```

Offset	Data	Source Line
0376	1670	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
0378	1670	CLS
037F	1670	GOSUB 7000 'PRINT SCREEN HEADING
0384	1670	
0384	1670	REM *****
0384	1670	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
0384	1670	REM *****
0384	1670	REM
0384	1670	OPEN DATFIL\$ FOR APPEND AS #1
0396	1670	
0396	1670	REM *****
0396	1670	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
0396	1670	REM *****
0396	1670	LPRINT CHR\$(15);
03A1	1670	WIDTH "LPT1:",160
03AB	1670	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
03B7	1672	COLOR 14
03BE	1672	LOCATE 11,26 : PRINT "OT REPEAT FORM "
03D3	1672	COLOR FORE,BACK,BORD
03E9	1672	
03E9	1672	REM *****
03E9	1672	REM **** COMMUNICATIONS SETUP ****
03E9	1672	REM *****
03E9	1672	REM PROTOCOL
03E9	1672	GOSUB 9001
03EE	1672	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0408	1676	
0408	1676	REM START SCANNER (S1)
0408	1676	CNTRLOPT =1 :GOSUB 9010
0414	1678	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
042E	1678	
042E	1678	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
0443	1678	READTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
044A	167A	
044A	167A	REM *****
044A	167A	REM **** SET SCAN SHEET CALL ****
044A	167A	REM *****
044A	167A	REM
044A	167A	
044A	167A	10 REM - RETURN POINT TO READ NEXT SHEET
044B	167A	
044B	167A	AS=INKEY\$
0454	167E	IF AS=CHR\$(27) THEN GOTO 25000
046A	167E	
046A	167E	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
046F	167E	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
048B	1682	
048B	1682	TEXT\$="" 'CLEAR THE INPUT AREA
0494	1682	GOSUB 8000 'DECODE HEADER
0499	1682	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
049E	1682	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
04A3	1682	LITHO\$ = MID\$(TEXT\$,22,8)
04C5	1686	GOSUB 8070 'CHECK FOR SCANNER ERRORS
04B1	1686	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

RACP910
OT REPEAT VISIT FORM

PAGE 7
04-26-87
09:33:04

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```
048F 1686
048F 1686 REM $INCLUDE: 'RACH910.MO1' INCLUDE THE REPEAT FORM REFORMAT/EDIT MOD
048F 1686 REM *****
048F 1686 REM **** AMBULATORY CARE INFORMATION SYSTEM 12 MAR 87 ****
048F 1686 REM **** D R BOLLING ****
048F 1686 REM **** MODULE NAME : RACH910.MO1 ****
048F 1686 REM **** SCANNER PROGRAM # : 910- OT REPEAT VISIT FORM ****
048F 1686 REM ****
048F 1686 REM **** PURPOSE : REFORMAT/EDIT THE ENCOUNTER ****
048F 1686 REM **** SHORT FORM OMR RECORD. ****
048F 1686 REM *****
048F 1686 REM **** RESERVED LINE NUMBERS 100-199 ****
048F 1686 REM *****
048F 1686
048F 1686 N.ERR =0 'COUNTS THE NUMBER OF ERRORS
04C6 1688
04C6 1688 REM *** LITHO CODE DONE IN BAS PROGRAM ***
04C6 1688
04C6 1688 REM *** CLINIC ID ***
04C6 1688 100 CL1.COD$="DHBA" 'DEFAULT CLINIC CODE
04CF 168C
04CF 168C REM *** SSN AND FMP ***
04CF 168C 102 SSN$=MID$(TEXT$,30,9)
04E1 1690 FMEMP$=MID$(TEXT$,39,2)
04F3 1694
04F3 1694 REM *** REFERRAL CODE ***
04F3 1694 104 X$=MID$(TEXT$,41,2) 'SINGLE BUBBLE CODE
0505 1698 IF X$=" " THEN 106
0513 1698 X=VAL(X$)
0520 169A REF.COD$=REFCOD$(X) 'USE TABLE
0532 169E GOTO 108
0536 169E
0536 169E 106 X=VAL(MID$(TEXT$,43,1))
054C 169E REF.COD$=CLINIC2.PFX$(X) + MID$(TEXT$,44,3)
056E 169E
056E 169E REM *** UNLISTED DX ***
056E 169E 108 X=VAL(MID$(TEXT$,47,1))
0584 169E IF X=0 THEN PRIMDX$=""
0598 16A2 IF X=1 THEN PRIMDX$="V"
05AC 16A2 IF X=2 THEN PRIMDX$="S"
05C0 16A2 X$=MID$(TEXT$,48,5)
05D2 16A2 IF X$=" " THEN 112
05E0 16A2
05E0 16A2 REM ... REMOVE LEADING BLANKS ...
05E0 16A2 110 IF LEFT$(X$,1)=" " THEN X$=RIGHT$(X$,4)+" ":GOTO 110
060F 16A2
060F 16A2 112 LASTC$=RIGHT$(X$,1) 'GET LAST CHAR
061E 16A6 IF LASTC$<>" " AND LASTC$<>"0" THEN PRIMDX$="" 'REMOVE V OR S
064A 16A6 PRIMDX$=LEFT$(PRIMDX$+X$,5) 'GET 5 CHAR ONLY
065C 16A6
065C 16A6 IF PRIMDX$<>" " THEN 113
066A 16A6 N.ERR=N.ERR+1
0672 16A6 ED.MSG$(N.ERR)="NO DIAGNOSIS CODED"
0686 16A6
```

Offset	Data	Source Line
0686	16A6	113 REM
0687	16A6	DAT.OFFSET = 53
068E	16A8	PFX.OFFSET = 57
0695	16AA	PR1.OFFSET = 58
069C	16AC	PRT.OFFSET = 62
06A3	16AE	P2X.OFFSET = 63
06AA	16B0	PR2.OFFSET = 64
0681	16B2	P2T.OFFSET = 68
0688	16B4	REA.OFFSET = 69
06BF	16B6	VCT.OFFSET = 70
06C6	16B8	UPR.OFFSET = 71
06CD	16BA	PRC.OFFSET = 76
06D4	16BC	PC2.OFFSET = 102
06DB	16BE	TOT.OFFSET = 75
06E2	16C0	POINTER = 0
06E9	16C2	
06E9	16C2	CH\$=MID\$(DATE\$,1,2) 'CURRENT MONTH
06FB	16C6	FOR I910 = 1 TO 4 'INITIALIZE VISIT DATE ARRAY
0702	16C6	VDATE\$(I910) = " "
0716	16C8	NEXT I910
0725	16C8	
0725	16C8	REM *** REPEAT THE FOLLOWING CODE 4 TIMES ***
0725	16C8	FOR I910= 1 TO 4
072C	16C8	
072C	16C8	REM *** VISIT DATE ***
072C	16C8	YR\$=MID\$(DATE\$,9,2) 'CURRENT YEAR
073E	16CC	X\$=MID\$(TEXT\$,DAT.OFFSET+POINTER,4) 'MONTH AND DAY FROM FORM
0755	16CC	IF X\$=" " THEN 199 'TIME TO QUIT
0763	16CC	IF LEFT\$(X\$,2)<=CH\$ THEN 116 'OK, USE THIS YEAR
0778	16CC	YR\$=RIGHT\$(STR\$(VAL(YR\$)-1),2) 'USE LAST YEAR
0798	16CC	116 VDATE\$ = YR\$ + X\$
07A6	16D0	'EDIT VISIT DATE
07A6	16D0	CK.5000\$=VDATE\$
07AF	16D4	GOSUB 5000 'DATE CHECK
07B4	16D4	CK.5010\$=VDATE\$
07BD	16D8	GOSUB 5010 'NUMERIC STRING CHECK
07C2	16D8	IF RT.5000 = 0 AND RT.5010 = 0 THEN GOTO 118
07E6	16DC	N.ERR=N.ERR+1
07EE	16DC	ED.MSG\$(N.ERR)="TODAYS DATE " + DATEERR\$(RT.5000)
0814	16DC	
0814	16DC	118 VDATE\$(I910)=VDATE\$
0828	16DC	
0828	16DC	REM *** PROVIDER 1 ID (PREFIX + NUM) ***
0829	16DC	120 PROV1.PFX\$ = MID\$(TEXT\$,PFX.OFFSET+POINTER,1)
083F	16E0	PROV1.NUM\$ = MID\$(TEXT\$,PR1.OFFSET+POINTER,4)
0856	16E4	PR1.COD\$(I910)=PROV1.PFX\$ + PROV1.NUM\$
086F	16E4	
086F	16E4	REM *** PRIMARY PROVIDER TIME ***
086F	16E4	121 X=VAL(MID\$(TEXT\$,PRT.OFFSET+POINTER,1))
088A	16E4	PR1.TIM\$(I910)=PROVIDER.TIMES(X)
08A7	16E4	
08A7	16E4	REM *** PROVIDER 2 ID (PFX + NUM) ***
08A7	16E4	122 PROV2.PFX\$ = MID\$(TEXT\$,P2X.OFFSET+POINTER,1)
08BE	16F8	PROV2.NUM\$ = MID\$(TEXT\$,PR2.OFFSET+POINTER,4)

PAGE 9
04-28-87
09:33:34

IBM Personal Computer BASIC Compiler v1.00

```

0805 16EC      PR2.COD$(1910)=PROV2.PFX$ + PROV2.NUM$
08EE 16EC      IF PR2.COD$(1910)=" " THEN PB=1 ELSE PB=0 'PROV2 BLANK IND
0917 16EE      IX$=STR$(1910)
0924 16F2      IF PROV2.PFX$=" " AND PROV2.NUM$<>" " THEN 124
094A 16F2      IF PROV2.PFX$<>" " AND PROV2.NUM$=" " THEN 124
0970 16F2      GOTO 125
0974 16F2
0974 16F2      124 N.ERR = N.ERR + 1
097C 16F2      ED.MSG$(N.ERR)="PROV 2 CODE MISSING PREFIX OR NUMBER IN PART"+IX$
0995 16F2
0995 16F2      REM *** SECONDARY PROVIDER TIME ***
0995 16F2      125 X=VAL(MID$(TEXT$,P2T.OFFSET+POINTER,1))
0980 16F2      PR2.TIM$(1910)=PROVIDER.TIM$(X)
09CD 16F2      REM IS THERE A TIME AND NO SEC PROV CODED?
09CD 16F2      IF PR2.TIM$(1910)<>"000" AND PB=1 THEN 126
09FA 16F2      REM IS THERE NO TIME AND A SEC PROV CODED?
09FA 16F2      IF PR2.TIM$(1910)="000" AND PB=0 THEN 127
0A27 16F2      GOTO 128
0A28 16F2
0A28 16F2      126 N.ERR = N.ERR + 1
0A33 16F2      ED.MSG$(N.ERR)="TIME CODED WITH NO SEC PROV CODED IN PART"+IX$
0A4C 16F2      GOTO 128
0A50 16F2
0A50 16F2      127 N.ERR = N.ERR + 1
0A58 16F2      ED.MSG$(N.ERR)="NO PROV 2 TIME IN PART"+IX$
0A71 16F2
0A71 16F2      REM *** REASON FOR #2 CARE PROVIDER ***
0A71 16F2      128 SPROV.REAS(1910)=MID$(TEXT$,REA.OFFSET+POINTER,1)
0A94 16F2      REM IS THER A REASON AND NO SEC PROV CODED?
0A94 16F2      IF SPROV.REAS(1910)<>" " AND PB=1 THEN 129
0AC1 16F2      REM IS THERE NO REASON AND A SEC PROV CODED?
0AC1 16F2      IF SPROV.REAS(1910)=" " AND PB=0 THEN 130
0AEE 16F2      GOTO 131
0AF2 16F2
0AF2 16F2      129 N.ERR = N.ERR + 1
0AFA 16F2      ED.MSG$(N.ERR)="REASON CODED WITH NO SEC PROV CODED IN PART"+IX$
0B13 16F2      GOTO 131
0B17 16F2
0B17 16F2      130 N.ERR = N.ERR + 1
0B1F 16F2      ED.MSG$(N.ERR)="NO PROV 2 REASON IN PART"+IX$
0B38 16F2
0B38 16F2      REM *** VISIT COUNT ***
0B38 16F2      131 VCNT$="1" 'DEFAULT
0B41 16F6      X$=MID$(TEXT$,VCT.OFFSET+POINTER,1)
0B58 16F6      IF X$="1" THEN VCNT$="0"
0B6F 16F6      IF X$>"1" THEN VCNT$=X$
0B86 16F6      VIS.CNT$(1910)=VCNT$
0B9A 16F6
0B9A 16F6      REM *** ADDITIONAL PROCEDURE
0B9A 16F6      132 ADDP$=MID$(TEXT$,UPR.OFFSET+POINTER,5)
0BB1 16FA
0BB1 16FA      REM *** PROCEDURE CODES FOR PROV 1 ***
0BB1 16FA      134 X$ = MID$(TEXT$,PRC.OFFSET+POINTER,N.PROC)
0BC9 16FA      GOSUB 6000 'DECODE THE X$ STRING

```

RACP910
OT REPEAT VISIT FORM

PAGE 10
04-28-87
09:33:34

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
08CE	16FA	IF TOT = 0 THEN GOTO 135
08D0	16FC	FOR 1910.A= 1 TO TOT
08EA	16FE	PTR=VAL(HOLDS(1910.A))
0C00	1702	GROUP1\$(1910,1910.A)=PROCEED\$(PTR)
0C22	1702	NEXT 1910.A
0C33	1702	REM ... ADD UNLISTED CODE IF THERE TO PROV 1 ...
0C33	1702	135 IF ADDPS=" " THEN 136
0C41	1702	TOT=TOT+1
0C49	1702	GROUP1\$(1910,TOT)=ADDPS
0C62	1702	
0C62	1702	136 TOT.PROC1(1910)=TOT
0C71	1702	
0C71	1702	REM *** PROCEDURE CODES FOR PROV 2 ***
0C71	1702	138 X\$ = MIDS(TEXTS,PC2.OFFSET+POINTER,N.PROC)
0C89	1702	GOSUB 6000 'DECODE THE X\$ STRING
0C8E	1702	IF TOT = 0 THEN GOTO 140
0C9D	1702	IF PB=1 THEN 141 'ERROR IF PROV 2 CODE BLANK
0CA8	1702	FOR 1910.A= 1 TO TOT
0CB5	1704	PTR=VAL(HOLDS(1910.A))
0CCB	1704	GROUP2\$(1910,1910.A)=PROCEED\$(PTR)
0CED	1704	NEXT 1910.A
0CFE	1704	
0CFE	1704	140 TOT.PROC2(1910)=TOT
0D0B	1704	GOTO 142
0D11	1704	
0D11	1704	141 N.ERR = N.ERR +1
0D19	1704	ED.MSG\$(N.ERR)="PROC FOR SEC PROV BUT NO SEC PROV CODED IN PART"+IX\$
0D32	1704	
0D32	1704	142 POINTER = POINTER + TOT.OFFSET
0D3D	1704	NEXT 1910
0D4F	1704	199 REM
0D5D	1704	
0D5D	1704	REM -----END OF MODULE RXXM910.M01-----
0D5D	1704	
0D5D	1704	IF N.ERR = 0 THEN GOTO 997
0D5F	1704	LPRINT "LITHO # ";LITHOS;" ... ERRORS"
0D71	1704	FOR 1997 = 1 TO N.ERR
0D7E	1706	LPRINT USING "### ";1997;
0D8A	1708	LPRINT "=> ";ED.MSG\$(1997)
0DA0	1708	NEXT 1997
0DB1	1708	LN.COUNT = LN.COUNT + N.ERR + 1
0DBD	170A	CNTRLOPT = 6
0DC4	170A	GOSUB 9010 'REJECT THE FORM
0DC9	170A	GOTO 998 'BYPASS THE DISK WRITER....
0DCD	170A	
0DD	170A	997 REM \$INCLUDE: 'RACH910.M02' REM INCLUDE THE BASE ENCOUNTER FORM DISK WRITER
0DCE	170A	REM *****
0DCF	170A	REM **** AMBULATORY CARE INFORMATION SYSTEM 13 MAR 87 ****
0DCE	170A	REM **** D R BOLLING ****
0DCE	170A	REM **** MODULE NAME : RACH910.M02 ****
0DCE	170A	REM **** SCANNER PROGRAM # : 910- OT REPEAT VISIT FORM ****
0DCE	170A	REM **** ****

RACP910
OT REPEAT VISIT FORM

PAGE 11
04-20-87
09:33:34

IBM Personal Computer BASIC Compiler V1.00

```
Offset Data Source Line
00CE 170A REM **** PURPOSE : CREATE AND WRITE THE DISK ****
00CE 170A REM **** RECORD FOR INPUT TO FOCUS ****
00CE 170A REM ****
00CE 170A REM **** PROGRAM ADDS PREFIX TO LITHO FOR EACH PATIENT ****
00CE 170A REM *****
00CE 170A REM **** RESERVED LINE NUMBERS 200-299 ****
00CE 170A REM *****
00CE 170A
00CE 170A REM BUILD THE OUTPUT RECORD
00CE 170A
00CE 170A REM *****
00CE 170A REM **** RECORD TYPE "1" - RECKEY PLUS TYPE 1 FIELDS ****
00CE 170A REM *****
00CE 170A REM RECOUNT$ ="9101"+RECKEY$ 'TRANSACTION IDENTIFIER
00CE 170A
00CE 170A REM ... TYPE 1 RECORD ...
00CE 170A
00CE 170A FOR IK = 1 TO 4
00D5 170A RECKEY$ = CL1.COD$ + VDATS(IK) + PR1.COD$(IK) + SSNS + FMEMPS
0E05 1710 IF VDATS(IK) = " " THEN 240
0E10 1710 RECOUNT$ = "9101" + RECKEY$ + RIGHTS(STR$(IK),1) + RIGHTS(LITHOS,7)
0E40 1714 TYPE1$ = "91" + VIS.CNT$(IK)+PR1.TIMS(IK)+PR2.COD$(IK)+PR2.TIMS(IK)
0E91 1718 TYPE1A$= SPROV.REAS(IK) + " " + REF.COD$ + STRING$(11," ") + PRIMDX$
0EC2 171C RECOUNT$ = RECOUNT$ + TYPE1$ + TYPE1A$
0ED5 171C GOSUB 280
0EDA 171C PRINT #1,RECOUNT$
0EE5 171C 240 NEXT IK
0EF7 171C
0EF7 171C REM *****
0EF7 171C REM **** RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE ****
0EF7 171C REM *****
0EF7 171C REM RECOUNT$ ="9102"+RECKEY$ 'TRANSACTION IDENTIFIER
0EF7 171C
0EF7 171C REM ... PROCEDURES OF FIRST PROVIDER ...
0EF7 171C
0EF7 171C FOR I910 = 1 TO 4
0EFE 171C FOR I910.A = 1 TO TOT.PROC1(I910)
0F12 171E RECKEY$ = CL1.COD$ + VDATS(I910) + PR1.COD$(I910) + SSNS + FMEMPS
0F42 171E IF VDATS(I910) = " " THEN 250
0F5A 171E RECOUNT$ = "9102" + RECKEY$ + RIGHTS(STR$(I910),1) + RIGHTS(LITHOS,7)
0F8A 171E RECOUNT$ =RECOUNT$ + "911" + GROUP1$(I910,I910.A)
0FAE 171E GOSUB 280
0FB3 171E PRINT #1,RECOUNT$
0FBE 171E NEXT I910.A
0FD2 171E 250 NEXT I910
0FE4 171E
0FE4 171E REM ... PROCEDURES OF SECOND PROVIDER ...
0FE4 171E
0FE4 171E FOR I910 = 1 TO 4
0FEB 171E FOR I910.A = 1 TO TOT.PROC2(I910)
0FFB 1720 RECKEY$ = CL1.COD$ + VDATS(I910) + PR1.COD$(I910) + SSNS + FMEMPS
102F 1720 IF VDATS(I910) = " " THEN 260
1047 1720 RECOUNT$ = "9102" + RECKEY$ + RIGHTS(STR$(I910),1) + RIGHTS(LITHOS,7)
1077 1720 RECOUNT$ =RECOUNT$ + "912" + GROUP2$(I910,I910.A)
```

RACP910
OT REPEAT VISIT FORM

PAGE 12
04-28-87
09:33:34

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
1098	1720	GOSUB 280
10A0	1720	PRINT #1,RECOUNTS
10AB	1720	NEXT I910.A
10BF	1720	260 NEXT I910
10D1	1720	
10D1	1720	GOTO 299
10D5	1720	
10D5	1720	REM *****
10D5	1720	REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
10D5	1720	REM *****
10D5	1720	280 PAD=MAXLENGTH - LEN(RECOUNTS) 'FIND OUT HOW SHORT THE RECORD IS
10E3	1722	RECOUNTS = RECOUNTS + STRING\$(PAD,PAD\$) 'PAD THE RECORD WITH FILL CHAR
10F6	1722	RETURN
10F9	1722	
10F9	1722	
10F9	1722	299 REM
10FA	1722	
10FA	1722	REM -----END OF MODULE RXXM910.MQ2-----
10FA	1722	
10FA	1722	998 REM CONTINUE
10FB	1722	
10FB	1722	999 READTYPE = 2
1102	1722	IF LN.COUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
1112	1722	GOTO 10
1116	1722	
1116	1722	REM END OF SCAN/DECODE/WRITE LOOP =====
1116	1722	
1116	1722	1000 REM \$INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
1117	1722	REM *****
1117	1722	REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
1117	1722	REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
1117	1722	REM *****
1117	1722	REM PATIENT OMR INPUT PROGRAM *
1117	1722	REM *
1117	1722	REM This program verifies user is logged on properly. If there is no *
1117	1722	REM valid user logged on at the time of execution, this subroutine will*
1117	1722	REM chain to the logon program RACP05, otherwise a return is issued. *
1117	1722	REM *****
1117	1722	REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
1117	1722	REM *****
1117	1722	1001 OPEN "I",1,"RACLOG.DAT"
1129	1722	IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
1137	1722	INPUT #1,USER\$(1),DTS,TMS,PIDS
1158	172E	IF USER\$(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
1166	172E	IF USER\$(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
1174	172E	CLOSE 1
1178	172E	SCREEN 0,1,0,0
1191	172E	COLOR FORE,BACK,BORD
11A7	172E	CLS
11A8	172E	RETURN
11AE	172E	
11AE	172E	1002 CLOSE
11 2	172E	CHAIN "RACP05"
1159	172E	'=====END OF LOGON VERIFY SUBROUTINE 1000=====

```

Offset  Data    Source Line
-----  -
11B9  172E
11B9  172E  2000  REM $INCLUDE: 'RACS2000.SUB' INCLUDE THE REPLY/DELAY SUB
11BA  172E  REM  *****
11BA  172E  REM  ****    AMBULATORY CARE DATA BASE          13 APR 85    ****
11BA  172E  REM  ****                                SKIP COLE      ****
11BA  172E  REM  ****    SUBROUTINE NAME      :    RACS2000.SUB          ****
11BA  172E  REM  ****    SCANNER PROGRAM #   :    ALL                  ****
11BA  172E  REM  ****    FUNCTION           :    THIS SUBROUTINE MODULE ****
11BA  172E  REM  ****                                SERVERS AS A WAIT AND REPLY ****
11BA  172E  REM  ****                                ENTRY MODULE      ****
11BA  172E  REM  ****    INPUT              :    SINGLE KEYBOARD ENTRY ****
11BA  172E  REM  ****                                ****
11BA  172E  REM  ****    OUTPUT              :    KEYBOARD ENTRY - UPPER CASE ****
11BA  172E  REM  ****                                ****
11BA  172E  REM  ****    RESERVED LINE                                ****
11BA  172E  REM  ****          NUMBERS      :    2001-2010          ****
11BA  172E  REM  *****
11BA  172E  2001  REM  REPLY FUNCTION
11B8  172E  2002  REPLY$=INKEY$ : IF REPLY$="" THEN 2002
11CF  1732      REPLY=ASC(REPLY$)
11D9  1734      IF REPLY > 90 THEN REPLY$=CHR$(REPLY XOR 32) 'CONVERT TO CAPS
11F4  1734      IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$="?"
1220  1734      RETURN
1223  1734
1223  1734  5000  REM $INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB
1224  1734  REM  *****
1224  1734  REM  ****    AMBULATORY CARE DATA BASE          13 APR 85    ****
1224  1734  REM  ****                                SKIP COLE      ****
1224  1734  REM  ****    SUBROUTINE NAME      :    RXXS5000.SUB          ****
1224  1734  REM  ****    SCANNER PROGRAM #   :    ALL                  ****
1224  1734  REM  ****    FUNCTION           :    THIS SUBROUTINE MODULE ****
1224  1734  REM  ****                                PERFORMS A DATE EDIT ****
1224  1734  REM  ****                                ****
1224  1734  REM  ****    INPUT              :    DATE TO BE CHECKED MUST BE ****
1224  1734  REM  ****                                IN THE VARIABLE NAMED ****
1224  1734  REM  ****                                'CK.5000$' ****
1224  1734  REM  ****                                IN THE FORMAT "YYMMDD" ****
1224  1734  REM  ****                                ****
1224  1734  REM  ****    OUTPUT              :    'RT.5000' IS THE RETURN CODE ****
1224  1734  REM  ****                                VARIABLE. IF THIS VARIABLE ****
1224  1734  REM  ****                                CONTAINS ANY NUMBER OTHER ****
1224  1734  REM  ****                                THAN 0, AN ERROR WAS FOUND ****
1224  1734  REM  ****                                IN THE DATE. ****
1224  1734  REM  ****                                ****
1224  1734  REM  ****    RESERVED LINE                                ****
1224  1734  REM  ****          NUMBERS      :    5001-5009          ****
1224  1734  REM  *****
1224  1734      RT.5000 = 0
1228  1734      CKYEAR = VAL(LEFT$(CK.5000$,2))    'YEAR NUMERIC VALUE
123E  1736      CKMONTH = VAL(MID$(CK.5000$,3,2)) 'MONTH NUMERIC VALUE
1254  1738      CKDAY = VAL(RIGHT$(CK.5000$,2))   'DAY NUMERIC VALUE
1267  173A
1267  173A      IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
127D  173A      IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009

```

```

Offset Data Source Line
1293 173A IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
12A9 173A IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
12BF 173A
12BF 173A REM LEAP YEAR CHECK
12BF 173A MOLENGTH(2) = 28
12C6 173A IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY
12D5 173A IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR
12EA 173A MOLENGTH(2) = 29
12F1 173A
12F1 173A 5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
1310 173A
1310 173A 5009 RETURN
1313 173A
1313 173A REM -----END OF SUBROUTINE 5000 -----
1313 173A
1313 173A 5010 REM $INCLUDE: 'RACS5010.SUB' INCLUDE THE NUMERIC STRING EDITOR
1314 173A REM *****
1314 173A REM **** AMBULATORY CARE DATA BASE 1 MAY 85 ****
1314 173A REM **** SKIP COLE ****
1314 173A REM **** SUBROUTINE NAME : RXXS5010.SUB ****
1314 173A REM **** SCANNER PROGRAM # : ALL ****
1314 173A REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1314 173A REM **** PERFORMS A NUMERIC STRING ****
1314 173A REM **** EDIT. ****
1314 173A REM ****
1314 173A REM **** INPUT : STRING TO BE EDITED IS IN ****
1314 173A REM **** THE VARIABLE NAMED ****
1314 173A REM **** 'CK.5010$' ****
1314 173A REM ****
1314 173A REM **** OUTPUT : 'RT.5010' IS THE RETURN CODE ****
1314 173A REM **** VARIABLE. IF THIS VARIABLE ****
1314 173A REM **** CONTAINS ANY NUMBER OTHER ****
1314 173A REM **** THAN 0, AN ERROR WAS FOUND ****
1314 173A REM **** IN THE STRING. ****
1314 173A REM ****
1314 173A REM **** RESERVED LINE ****
1314 173A REM **** NUMBERS : 5011-5019 ****
1314 173A REM *****
1314 173A REM RT.5010 = 0
1318 173A
1318 173A FOR I5010 = 1 TO LEN(CK.5010$)
1328 173C J5010= ASC(MID$(CK.5010$,I5010,1))
133F 1740 IF J5010 < 48 OR J5010 > 57 THEN RT.5010 = RT.5010 + 1
1367 1740 NEXT I5010
1378 1740
1378 1740 RETURN
1378 1740 REM ----- END OF SUBROUTINE 5010 -----
1378 1740 5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTPATIENT UCA CHECK SUB
137C 1740
137C 1740 6000 REM $INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE ROUTINE
137D 1740 REM *****
137D 1740 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
137D 1740 REM **** SKIP COLE ****
137D 1740 REM **** SUBROUTINE NAME : RXXS6000.SUB ****

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

137D 1740 REM **** SCANNER PROGRAM # : ALL ****
137D 1740 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
137D 1740 REM **** PERFORMS INSTRING SEARCH ****
137D 1740 REM ****
137D 1740 REM **** INPUT : STRING TO BE SEARCHED MUST ****
137D 1740 REM **** BE IN THE VARIABLE NAMED : ****
137D 1740 REM **** 'XS' ****
137D 1740 REM ****
137D 1740 REM **** OUTPUT : 'TOT' = TOTAL NUMBER OF ****
137D 1740 REM **** HITS IN THE DESTING ****
137D 1740 REM **** 'HOLD$( )' IS THE ARRAY ****
137D 1740 REM **** CONTAINING THE NUMERIC ****
137D 1740 REM **** VALUE OF THE HIT POSITIONS ****
137D 1740 REM ****
137D 1740 REM *****
137D 1740 REM **** RESERVED LINE ****
137D 1740 REM **** NUMBERS : 6001-6009 ****
137D 1740 REM *****
137D 1740 6001 PTR = INSTR(X$, "1")
138B 1740 TOT = 0
1392 1740 WHILE PTR > 0
139D 1740 TOT=TOT+1
13A5 1740 HOLD$(TOT) = RIGHT$(STR$(PTR),2)
13C7 1740 PTR=PTR+1
13CF 1740 PTR = INSTR(PTR,X$, "1")
13E1 1740 WEND
13E5 1740 RETURN
13E8 1740 REM -----END OF SUBROUTINE RXXS6000.SUB-----
13E8 1740
13E8 1740 7000 REM $INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
13E9 1740 REM *****
13E9 1740 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
13E9 1740 REM **** SKIP COLE ****
13E9 1740 REM **** SUBROUTINE NAME : RACS7000.SUB ****
13E9 1740 REM **** SCANNER PROGRAM # : ALL ****
13E9 1740 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
13E9 1740 REM **** PRINTS THE STANDARD SCREEN ****
13E9 1740 REM **** HEADING. ****
13E9 1740 REM **** INPUT : COMMON VARIABLE USERS(2) ****
13E9 1740 REM **** SYSTEM DATE ****
13E9 1740 REM ****
13E9 1740 REM **** OUTPUT : SCREEN HEADING ****
13E9 1740 REM ****
13E9 1740 REM **** RESERVED LINE ****
13E9 1740 REM **** NUMBERS : 7001-7010 ****
13E9 1740 REM *****
13E9 1740
13E9 1740 7001 LOCATE 1,1
13F3 1740 PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
13FB 1740 LOCATE 1,65
1408 1740 PRINT DATES;
1410 1740 LOCATE 2,1

```

Offset Data Source Line

```

141D 1740      PRINT "USER : ";USER$(1)
142A 1740      RETURN
142D 1740 7100  REM $INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
142E 1740 REM *****
142E 1740 REM *****  AMBULATORY CARE DATA BASE          13 APR 85  *****
142E 1740 REM *****                                SKIP COLE *****
142E 1740 REM *****  SUBROUTINE NAME      :  RXXS7100.SUB *****
142E 1740 REM *****  SCANNER PROGRAM #    :  ALL *****
142E 1740 REM *****  FUNCTION              :  THIS SUBROUTINE MODULE *****
142E 1740 REM *****                                PRINTS THE STANDARD HEADING *****
142E 1740 REM *****                                ON THE PRINTER. *****
142E 1740 REM *****  INPUT                  :  DATE,PAGE,PGMID$,PGMTITL$ *****
142E 1740 REM *****                                *****
142E 1740 REM *****  OUTPUT                  :  PRINTER HEADING, LN.COUNT *****
142E 1740 REM *****                                *****
142E 1740 REM *****  RESERVED LINE *****
142E 1740 REM *****      NUMBERS      :  7101-7110 *****
142E 1740 REM *****
142E 1740 7101  IF PAGE > 0 THEN LPRINT CHR$(12);
1444 1740      LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITL$;
1451 1740      LPRINT TAB(70);DATE$
1464 1740      PAGE=PAGE+1
146C 1740      LPRINT "PROGRAM ";PGMID$;TAB(70);"PAGE";
1489 1740      LPRINT USING "####";PAGE
1495 1740      LPRINT
149D 1740      LN.COUNT=3
14A4 1740      RETURN
14A7 1740 8000  REM $INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
14AB 1740 REM *****
14AB 1740 REM *****  AMBULATORY CARE DATA BASE          13 APR 85  ***
14AB 1740 REM *****                                SKIP COLE ***
14AB 1740 REM *****  SUBROUTINE NAME      :  RXXS8000.SUB ***
14AB 1740 REM *****  SCANNER PROGRAM #    :  ALL ***
14AB 1740 REM *****  FUNCTION              :  THIS SUBROUTINE MODULE ***
14AB 1740 REM *****                                IS A GROUPING THAT PERFORMS ***
14AB 1740 REM *****                                VARIOUS DECODING FUNCTIONS ***
14AB 1740 REM *****                                ON THE SCANNER DATA ***
14AB 1740 REM *****
14AB 1740 REM *****  8001  - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
14AB 1740 REM *****  8050  - CHECK FOR END OF JOB ***
14AB 1740 REM *****  8100  - PRINT THE HEADER DATA ON THE SCREEN ***
14AB 1740 REM *****  8200  - DECODE THE RESPONSE POSITIONS (POINTER 21-..) ***
14AB 1740 REM *****                                (RETURNED IN TEXT$ STRING VARIABLE) ***
14AB 1740 REM *****
14AB 1740 REM *****  INPUT                  :  SHEET RECORD, RECORD LENGTH ***
14AB 1740 REM *****
14AB 1740 REM *****  OUTPUT                  :  'TEXT$' TRING VARIABLE ***
14AB 1740 REM *****
14AB 1740 REM *****  RESERVED LINE *****
14AB 1740 REM *****      NUMBERS      :  8001-8500 ***
14AB 1740 REM *****

```

Offset Data Source Line

```

14A8 1740 'DECODE THE HEADER ONLY
14A8 1740 8001     POINTER = 0
14AF 1740         RECORDPTR = VARPTR(SHEETREC(0))
14B6 1742         FOR J8000 = 1 TO 21
14BD 1742 8002     TEXT$= TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
14DB 1742         POINTER=POINTER+1
14E3 1742         NEXT J8000
14F2 1744         PROGRAM$= LEFT$(TEXT$,3)
1501 1748         BATCH$=  MID$(TEXT$,4,3)
1513 174C         SERIAL$= MID$(TEXT$,7,4)
1525 1750         RUNID$=  MID$(TEXT$,11,1)
1537 1754         FORM$=   MID$(TEXT$,12,2)
1549 1758         POCKET$= MID$(TEXT$,14,1)
155B 175C         SCANERR1$=MID$(TEXT$,16,2)
156D 1760         SCANERR2$=MID$(TEXT$,18,2)
157F 1764         SCANERR3$=MID$(TEXT$,20,2)
1591 1768         GOTO 8500
1595 1768
1595 1768 8050 REM CHECK FOR END OF JOB/END OF BATCH
1596 1768         IF PROGRAM$ = PGMIID$ THEN GOTO 8500
15A8 1768         LPRINT STRING$(80,"")
15B6 1768         LPRINT
15BE 1768         LPRINT "RECORDS PROCESSED ... ";SERIAL$
15CB 1768         LPRINT "STARTED AT ..... ";BTIM$
15DB 1768         LPRINT "ENDED   AT ..... ";TIM$
15E5 1768         LPRINT CHR$(12)
15F0 1768         GOTO 30000
15F4 1768
15F4 1768 8070 REM CHECK FOR SCANNER ERRORS
15F5 1768         IF POCKET$ = " " GOTO 8500
1607 1768         LPRINT LITHOS;
160F 1768         LPRINT " ... SCANNER ERRORS : ";
1617 1768         LPRINT SCANERR1$;" / ";
1624 1768         LPRINT SCANERR2$;" / ";
1631 1768         LPRINT SCANERR3$
1639 1768         LN=LN+1
1641 176A         GOTO 999
1645 176A
1645 176A 8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
1646 176A         LOCATE 5,1:PRINT "PROGRAM ";PROGRAM$;
1660 176A
1660 176A         PRINT " BATCH ";BATCH$;
166D 176A         PRINT "  RUN ";RUNID$;
167A 176A         PRINT "  FORM ";FORM$;
1687 176A         PRINT " POCKET ";POCKET$
1694 176A         GOTO 8500
1698 176A
1698 176A 8200 REM DECODE THE RESPONSE POSITIONS
1699 176A         POINTER = 21
16A0 176A         RECORDPTR = VARPTR(SHEETREC(0))
16A7 176A         FOR J8000 = 22 TO RECORDLENGTH
16B4 176C 8202     TEXT$ = TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
16D2 176C         POINTER=POINTER+1
16DA 176C         NEXT J8000

```

Offset	Data	Source Line
16EB	176C	
16EB	176C	8500 RETURN
16EE	176C	
16EE	176C	REM ----- END OF RXXS8000.SUB -----
16EE	176C	
16EE	176C	9000 REM \$INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
16EF	176C	REM *****
16EF	176C	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
16EF	176C	REM **** SKIP COLE ****
16EF	176C	REM **** PROGRAM NAME : RACS9000.SUB ****
16EF	176C	REM **** SCANNER PROGRAM # : ALL ****
16EF	176C	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
16EF	176C	REM **** CONTROLS THE SCANNER I/O ****
16EF	176C	REM ****
16EF	176C	REM **** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS ****
16EF	176C	REM **** COMMUNICATIONS MANUAL AND THE ****
16EF	176C	REM **** PRE-RELEASED SOFTWARE GUIDE ****
16EF	176C	REM ****
16EF	176C	REM *****
16EF	176C	REM **** RESERVED LINE ****
16EF	176C	REM **** NUMBERS : 9001-9100 ****
16EF	176C	REM *****
16EF	176C	
16EF	176C	REM *****
16EF	176C	REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
16EF	176C	REM **** ARGUMENTS: PRESET ... SEE BELOW ****
16EF	176C	REM *****
16EF	176C	9001 REM
16F0	176C	PROTOCOL(0) = 9600 'BAUD RATE
16F7	176C	PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
16FE	176C	PROTOCOL(2) = 8 'DATA BITS
1705	176C	PROTOCOL(3) = 1 'STOP BITS
170C	176C	PROTOCOL(4) = 2 'RS-232 PORT
1713	176C	PROTOCOL(5) = 0 'WRITE TIME-OUT
171A	176C	PROTOCOL(6) = 0 'READ TIME-OUT
1721	176C	
1721	176C	ERRSTAT\$ = SPACES(60)
172D	176C	ARGPTR = VARPTR(PROTOCOL(0))
1734	176E	CALL SETUP (ARGPTR,ERRSTAT\$)
1745	176E	ERRMSG\$=""
174E	176E	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="SETUP ERROR "+ERRSTAT\$
176A	176E	GOTO 9100
176E	176E	
176E	176E	REM *****
176E	176E	REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
176E	176E	REM **** ARGUMENTS: CNTRLOPT ****
176E	176E	REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
176E	176E	REM **** CNTRLOPT = 2 = STOP SCANNER (S0) ****
176E	176E	REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
176E	176E	REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
176E	176E	REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
176E	176E	REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
176E	176E	REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****

RACP910
OT REPEAT VISIT FORM

PAGE 19
04-28-87
09:33:34

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
176E	176E	REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
176E	176E	REM *****
176E	176E	9010 REM
176F	176E	ERRSTAT\$ = SPACES(60)
177B	176E	CALL CNTRLO (CNTRLOPT,ERRSTAT\$)
178C	176E	ERRMSG\$=""
1795	176E	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
17B1	176E	GOTO 9100
1785	176E	
1785	176E	REM *****
1785	176E	REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
1785	176E	REM ****
1785	176E	REM **** ARGUMENTS: READTYPE ****
1785	176E	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
1785	176E	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
1785	176E	REM ****
1785	176E	REM **** ARGUMENTS: RECORDLENGTH ****
1785	176E	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
1785	176E	REM **** TRANSMITTED ****
1785	176E	REM *****
1785	176E	9020 REM
1786	176E	ERRSTAT\$ = SPACES(60)
17C2	176E	RECORDPTR = VARPTR(SHEETREC(0))
17C9	176E	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
17E2	176E	ERRMSG\$=""
17EB	176E	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
180C	176E	GOTO 9100
1810	176E	
1810	176E	REM *****
1810	176E	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
1810	176E	REM ****
1810	176E	REM **** ARGUMENTS: PRINTPOS ****
1810	176E	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
1810	176E	REM **** VALUES = 0 THRU 90 ****
1810	176E	REM ****
1810	176E	REM **** ARGUMENTS: PSTRINGS ****
1810	176E	REM **** TEXT TO BE PRINTED ON THE FORM ****
1810	176E	REM ****
1810	176E	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
1810	176E	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
1810	176E	REM *****
1810	176E	9030 REM
1811	176E	ERRSTAT\$ = SPACES(60)
181D	176E	RECORDPTR = VARPTR(SHEETREC(0))
1824	176E	CALL TPRINT(PRINTPOS,PSTRINGS\$,ERRSTAT\$)
1839	1774	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
1855	1774	GOTO 9100
1859	1774	
1859	1774	9100 RETURN
185C	1774	REM -----END OF SUBROUTINE RACS9000.SUB -----
185C	1774	
185C	1774	REM END OF SUBROUTINES =====
185C	1774	
185C	1774	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!

RACP910
OT REPEAT VISIT FORM

PAGE 20
04-28-87
09:33:34

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

185D	1774	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1865	1774	LPRINT "ERASING FILE ";DATFIL\$
1872	1774	BEEP
1876	1774	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1882	1774	CLOSE
1886	1774	OPEN DATFIL\$ FOR OUTPUT AS #1
1898	1774	PRINT #1,STRINGS(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
18AA	1774	CLOSE
18AE	1774	
18AE	1774	30000 REM
18AF	1774	CLOSE
1883	1774	CHAIN "RACP10"
188A	1774	END
188E	1774	
18C1	1774	

22151 Bytes Available
15807 Bytes Free

0 Warning Error(s)
0 Severe Error(s)

PAGE 1

01-28-87

09:26:25

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

001A 0002 REM \$LINESIZE: 132

001A 0002 REM \$PAGESIZE: 66

001A 0002 REM \$TITLE: 'RACP920 '

001A 0002 REM \$SUBTITLE: 'PT REPEAT VISIT FORM'

001A 0002 REM \$PAGE

RACP920
PT REPEAT VISIT FORM

PAGE 2
04-28-87
09:28:25

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
001A 0002	REM	-----
001A 0002	REM	NAME: RACP910 AMBULATORY CARE INFORMATION SYSTEM
001A 0002	REM	DATE: 13 MAR 87 PT (DHDA) REPEAT VISIT
001A 0002	REM	D R BOLLING SHORT FORM
001A 0002	REM	
001A 0002	REM	INCLUDES PREFIX TO LITHO FOR EACH PATIENT ON FORM
001A 0002	REM	-----
001A 0002	REM	PT REPEAT FORM INPUT PROGRAM
001A 0002	REM	
001A 0002	REM	This program reads the SHORT form OMR data, converts various
001A 0002	REM	fields, prints an error report and produces the file:
001A 0002	REM	
001A 0002	REM	VISIT.DAT
001A 0002	REM	
001A 0002	REM	for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002	REM	each time the program is run. Thus, if the file does not exist,
001A 0002	REM	records will be added to the front. If the file exists, records
001A 0002	REM	will be added to the end of the current file. It is intended that
001A 0002	REM	the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002	REM	the data file after the load has been successfully accomplished.
001A 0002	REM	
001A 0002	REM	
001A 0002	REM	If there is no valid user logged at the time of execution, this
001A 0002	REM	program will chain to the logon program RXXPO5, otherwise,
001A 0002	REM	THE PROGRAM CHAINS TO PROGRAM RACP10 ON EXIT.
001A 0002	REM	
001A 0002	REM	\$INCLUDE: 'RACDIM.MOD' REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002	REM	*****
001A 0002	REM	* NAME: RACDIM.MOD DIMENSION DEFINITIONS *
001A 0002	REM	* Date: 28 Feb 84 Written by: Floyd Cole *
001A 0002	REM	*****
001A 0002	REM	' Dimensioned variables are defined in this file.
001A 0002	REM	' It is an included file so it cannot be run in a stand-alone,
001A 0002	REM	' mode.
001A 0002	REM	'
001A 0002	REM	' This program segment may be modified, but all files containing
001A 0002	REM	' an include for this segment must be re-compiled in order to
001A 0002	REM	' affect the changes made here.
001A 0002	REM	' ***** START OF DIMENSION DEFINITION *****
001A 0002	REM	
001A 0002	REM	DEFINT A-Z
001A 0002	REM	DIM USERS\$(2),MOLENGTH(12),DATEERR\$(3)
001A 0002	REM	
001A 0002	REM	' ***** END OF DIMENSION DEFINITIONS *****
001A 0002	REM	
001A 0002	REM	DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002	REM	
001A 0002	REM	DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002	REM	DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002	REM	DIM ED.MSG\$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002	REM	DIM CLINIC2.PFX\$(6) '(PREFIX -A B C D F S- FOR CLINIC #2)
001A 0002	REM	DIM PROCED\$(26) '(PROCEDURE TABLE FOR SHORT FORM)
001A 0002	REM	DIM PROVIDER.TIMES(8) '(PROVIDER TIME TABLE)

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

001A 0002      DIM REFCOD$(18)      '(REFERRAL CODE TABLE)
001A 0002      DIM VDAT$(4)        '
001A 0002      DIM PR1.COD$(4)      '
001A 0002      DIM PR1.TIMS(4)      '   ARRAYS TO HOLD MULTIPLE VISITS
001A 0002      DIM PR2.COD$(4)      '
001A 0002      DIM PR2.TIMS(4)      '
001A 0002      DIM SPROV.REAS(4)    '
001A 0002      DIM VIS.CNT$(4)      '
001A 0002      DIM TOT.PROC1(4)     '
001A 0002      DIM TOT.PROC2(4)     '
001A 0002      DIM GROUP1$(6,27)    '(PATIENT/PROCEDURE GROUP PROV 1)
001A 0002      DIM GROUP2$(6,26)    '(PATIENT/PROCEDURE GROUP PROV 2)
001A 0002      DIM HOLDS(26)        '(HOLD AREA FOR SUBROUTINE 6000)
001A 0002
001A 0002      REM $INCLUDE: 'RACCMN.MOD'      REM INCLUDE THE COMMON AREA DEFINITION
001A 0002      *****
001A 0002      '*      NAME: RACCMN.MOD              COMMON AREA DEFINITION      *
001A 0002      '*      Date: 28 Feb 84              Written by: Floyd Cole      *
001A 0002      *****
001A 0002      '   COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE.  IT IS AN
001A 0002      '   INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE,  MODE.
001A 0002      '
001A 0002      '   This program segment may be modified, but all files containing
001A 0002      '   an include for this segment must be re*compiled in order to
001A 0002      '   affect the changes made here.
001A 0002      '
001A 0002      '   *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002      COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL$ 'BASIC SCREEN COLORS
001A 0002      COMMON HEADERS$      '21 CHARACTER SCANNER HEADER INFO
001A 0002      COMMON TEXT$        '  AINING CHARACTERS FROM SCANNER
001A 0002      COMMON PGMIDS$      'PROGRAM OR FORM ID
001A 0002      COMMON MOLENGTH( )  'DAYS IN THE MONTH
001A 0002      COMMON USERS( )
001A 0002      '   *****END   OF COMMON DEFINITION*****
001A 0002
001A 0002
001A 0002      REM $INCLUDE: 'RACDEF.MOD'      REM INCLUDE THE DEFAULT DEFINITIONS
001A 0002      *****
001A 0002      '*      NAME: RACP01.DEF              DEFAULT DEFINITIONS      *
001A 0002      '*      Date: 28 Feb 84              Written by: Floyd Cole      *
001A 0002      *****
001A 0002      '   Variables used in common that have a default value on start*up
001A 0002      '   will be held in this file.  It is an included file so it cannot
001A 0002      '   be run in a stand*alone mode.  In normal operation, this file
001A 0002      '   should be 'included' in the main program only (RACP10.BAS).
001A 0002      '
001A 0002      '   This program segment may be modified, but all files containing
001A 0002      '   an include for this segment must be re*compiled in order to
001A 0002      '   affect the changes made here.
001A 0002      '
001A 0002      '   *****START OF DEFAULT DEFINITION*****
001A 0002
001A 0002      FORE = 15      'FOREGROUND COLOR = INTENSE WHITE

```

Offset Data Source Line

```

0050 1654      BACK = 1      'Background Color = Light Blue
0057 1654      BORD = 4      'BORDER          = RED
005E 1656      HIDE = 4      'ALTERNATE COLOR = RED
0065 1656      EFORE= 14     'ERROR FOREGROUND DISPLAY
006C 1656      EBACK= 0      'ERROR BACKGROUND DISPLAY
0073 1656      BELLS = CHR$(7) 'Sound the bell
007F 1656
007F 1656      MOLENGTH(1) = 31      'JAN
0086 1656      MOLENGTH(2) = 28      'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
008D 1656      MOLENGTH(3) = 31      'MAR
0094 1656      MOLENGTH(4) = 30      'APR
009B 1656      MOLENGTH(5) = 31      'MAY
00A2 1656      MOLENGTH(6) = 30      'JUN
00A9 1656      MOLENGTH(7) = 31      'JUL
00B0 1656      MOLENGTH(8) = 31      'AUG
00B7 1656      MOLENGTH(9) = 30      'SEP
00BE 1656      MOLENGTH(10) = 31     'OCT
00C5 1656      MOLENGTH(11) = 30     'NOV
00CC 1656      MOLENGTH(12) = 31     'DEC
00D3 1656
00D3 1656      DATEERR$(0) = " "
00DC 1656      DATEERR$(1) = "INVALID MONTH"
00E5 1656      DATEERR$(2) = "INVALID DAY  "
00EE 1656      DATEERR$(3) = "DAY TOO LARGE FOR MONTH CODED"
00F7 1656
00F7 1656      MAXLENGTH = 80      'MAXIMUM LENGTH OF OUTPUT RECORD
00FE 1658      PAD$ = "."      'PAD CHARACTER FOR SHORT RECORDS
0107 165C
0107 165C      ' *****END OF DEFAULT DEFINITION*****
0107 165C
0107 165C      KEY OFF
010D 165C
010D 165C      REM *****
010D 165C      REM THE FOLLOWING VARIABLES ARE UNIQUE TO EACH PROGRAM AND MUST
010D 165C      REM BE CHANGED.
010D 165C      REM *****
010D 165C      PGMTITL$ = "PT (DHDA) REPEAT FORM"
0116 1660
0116 1660      PGMIDS = "920"      'VALUE RECEIVED FROM THE SCANNER
011F 1660      'IN HEADER VARIABLE 'PROGRAM$'
011F 1660
011F 1660      DATFIL$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
0128 1664
0128 1664      REM LENGTH OF STRING RECEIVED FROM THE OMR....
0128 1664      HEADER = 21
012F 1666      RESPONSE= 331
0136 1668      RECORDLENGTH = HEADER + RESPONSE
0141 166A
0141 166A      N.PROC = 26      ' NUMBER OF PROCEDURES FOR THIS FORM
0148 166C
0148 166C      REM *****
0148 166C
0148 166C      BTIME$=TIME$      'SCAN START TIME

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

0151 1670
0151 1670
0151 1670 REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0151 1670 REM CLINIC #2
0151 1670 CLINIC2.PFX$(0)=" "
015A 1670 CLINIC2.PFX$(1)="A"
0163 1670 CLINIC2.PFX$(2)="B"
016C 1670 CLINIC2.PFX$(3)="C"
0175 1670 CLINIC2.PFX$(4)="D"
017E 1670 CLINIC2.PFX$(5)="F"
0187 1670 CLINIC2.PFX$(6)="S"
0190 1670
0190 1670 REM *** PREASSIGNED REFERRAL CODES ***
0190 1670 REFCOD$(1)="AAAA" : REFCOD$(07)="BEAA" : REFCOD$(13)="BHAF"
01A8 1670 REFCOD$(2)="AEAA" : REFCOD$(08)="BEDA" : REFCOD$(14)="BHAG"
01C6 1670 REFCOD$(3)="BAAA" : REFCOD$(09)="BEFA" : REFCOD$(15)="BHAH"
01E1 1670 REFCOD$(4)="BAKA" : REFCOD$(10)="BGYA" : REFCOD$(16)="BHA1"
01FC 1670 REFCOD$(5)="BBAA" : REFCOD$(11)="BHAA" : REFCOD$(17)="BIYA"
0217 1670 REFCOD$(6)="BBCA" : REFCOD$(12)="BHAE" : REFCOD$(18)="DHCA"
0232 1670
0232 1670 REM *** ENCOUNTER FORM PROCEDURE TABLE ***
0232 1670 PROCED$(0)=" " : PROCED$(10) = "06027" : PROCED$(20) = "04014"
0240 1670 PROCED$(1)="06030" : PROCED$(11) = "06013" : PROCED$(21) = "06017"
0268 1670 PROCED$(2)="97118" : PROCED$(12) = "06014" : PROCED$(22) = "06023"
0283 1670 PROCED$(3)="97114" : PROCED$(13) = "04021" : PROCED$(23) = "97012"
029E 1670 PROCED$(4)="97116" : PROCED$(14) = "06034" : PROCED$(24) = "97128"
02B9 1670 PROCED$(5)="06005" : PROCED$(15) = "06035" : PROCED$(25) = "97022"
02D4 1670 PROCED$(6)="97010" : PROCED$(16) = "06063" : PROCED$(26) = "12008"
02EF 1670 PROCED$(7)="06011" : PROCED$(17) = "04011"
0301 1670 PROCED$(8)="06083" : PROCED$(18) = "04020"
0313 1670 PROCED$(9)="06028" : PROCED$(19) = "97024"
0325 1670
0325 1670 REM *** PROVIDER TIME TABLE ***
0325 1670 PROVIDER.TIMES$(00)="000" ' NO TIME
032E 1670 PROVIDER.TIMES$(01)="005" ' 5 MINUTES
0337 1670 PROVIDER.TIMES$(02)="010" '10 MINUTES
0340 1670 PROVIDER.TIMES$(03)="015" '15 MINUTES
0349 1670 PROVIDER.TIMES$(04)="020" '20 MINUTES
0352 1670 PROVIDER.TIMES$(05)="030" '30 MINUTES
035B 1670 PROVIDER.TIMES$(06)="045" '45 MINUTES
0364 1670 PROVIDER.TIMES$(07)="060" '60 MINUTES
036D 1670 PROVIDER.TIMES$(08)="999" ' > 1 HOUR
0376 1670
0376 1670 REM INCLUDE: 'UCABAMC.OPT' INCLUDE OUTPATIENT UCA TABLE
0376 1670
0376 1670 REM $PAGE

```

RACP920
PT REPEAT VISIT FORM

PAGE 6
04-28-87
09:28:25

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
0376	1670	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
037B	1670	CLS
037F	1670	GOSUB 7000 'PRINT SCREEN HEADING
0384	1670	
0384	1670	REM *****
0384	1670	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
0384	1670	REM *****
0384	1670	REM
0384	1670	OPEN DATFIL\$ FOR APPEND AS #1
0396	1670	
0396	1670	REM *****
0396	1670	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
0396	1670	REM *****
0396	1670	LPRINT CHR\$(15);
03A1	1670	WIDTH "LPT1:",160
03AB	1670	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
03B7	1672	COLOR 14
03BE	1672	LOCATE 11,26 : PRINT "PT REPEAT FORM "
03D3	1672	COLOR FORE,BACK,BORD
03E9	1672	
03E9	1672	REM *****
03E9	1672	REM **** COMMUNICATIONS SETUP ****
03E9	1672	REM *****
03E9	1672	REM PROTOCOL
03E9	1672	GOSUB 9001
03EE	1672	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0408	1676	
0408	1676	REM START SCANNER (S1)
0408	1676	CNTRLOPT =1 :GOSUB 9010
0414	1678	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
042E	1678	
042E	1678	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
0443	1678	READYTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
044A	167A	
044A	167A	REM *****
044A	167A	REM **** SET SCAN SHEET CALL ****
044A	167A	REM *****
044A	167A	REM
044A	167A	
044A	167A	10 REM - RETURN POINT TO READ NEXT SHEET
044B	167A	
044B	167A	AS=INKEY\$
0454	167E	IF AS=CHR\$(27) THEN GOTO 25000
046A	167E	
046A	167E	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
046F	167E	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
048B	1682	
048B	1682	TEXT\$="" 'CLEAR THE INPUT AREA
0494	1682	GOSUB 8000 'DECODE HEADER
0499	1682	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
049E	1682	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
04A3	1682	LITHO\$ = MID\$(TEXT\$,22,8)
04B5	1686	GOSUB 8070 'CHECK FOR SCANNER ERRORS
04BA	1686	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

RACP920
PT REPEAT VISIT FORM

PAGE 7
04-28-87
09:26:25

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
04BF	1686	
04BF	1686	REM \$INCLUDE: 'RACH920.M01' INCLUDE THE SHORT FORM REFORMAT/EDIT MOD
04BF	1686	REM *****
04BF	1686	REM **** AMBULATORY CARE INFORMATION SYSTEM 13 MAR 87 ****
04BF	1686	REM **** D R BOLLING ****
04BF	1686	REM **** MODULE NAME : RACH920.M01 ****
04BF	1686	REM **** SCANNER PROGRAM # : 920- PT REPEAT VISIT FORM ****
04BF	1686	REM ****
04BF	1686	REM **** PURPOSE : REFORMAT/EDIT THE ENCOUNTER ****
04BF	1686	REM **** SHORT FORM OMR RECORD. ****
04BF	1686	REM *****
04BF	1686	REM **** RESERVED LINE NUMBERS 100-199 ****
04BF	1686	REM *****
04BF	1686	
04BF	1686	N.ERR =0 'COUNTS THE NUMBER OF ERRORS
04C6	1688	
04C6	1688	REM *** LITHO CODE DONE IN BAS PROGRAM ***
04C6	1688	
04C6	1688	REM *** CLINIC ID ***
04C6	1688	100 CL1.COD\$="DHDA" 'DEFAULT CLINIC CODE
04CF	168C	
04CF	168C	REM *** SSN AND FMP ***
04CF	168C	102 SSN\$=MID\$(TEXT\$,30,9)
04E1	1690	FMEMP\$=MID\$(TEXT\$,39,2)
04F3	1694	
04F3	1694	REM *** REFERRAL CODE ***
04F3	1694	104 X\$=MID\$(TEXT\$,41,2) 'SINGLE BUBBLE CODE
0505	1698	IF X\$=" " THEN 106
0513	1698	X=VAL(X\$)
0520	169A	REF.COD\$=REFCOD\$(X) 'USE TABLE
0532	169E	GOTO 108
0536	169E	
0536	169E	106 X=VAL(MID\$(TEXT\$,43,1))
054C	169E	REF.COD\$=CLINIC2.PFX\$(X) + MID\$(TEXT\$,44,3)
056E	169E	
056E	169E	REM *** UNLISTED DX ***
056E	169E	108 X=VAL(MID\$(TEXT\$,47,1))
0584	169E	IF X=0 THEN PRIMDX\$=""
0598	16A2	IF X=1 THEN PRIMDX\$="V"
05AC	16A2	IF X=2 THEN PRIMDX\$="S"
05C0	16A2	X\$=MID\$(TEXT\$,48,5)
05D2	16A2	IF X\$=" " THEN 112
05E0	16A2	
05E0	16A2	REM ... REMOVE LEADING BLANKS ...
05E0	16A2	110 IF LEFT\$(X\$,1)=" " THEN X\$=RIGHT\$(X\$,4)+" ":GOTO 110
060F	16A2	
060F	16A2	112 LASTC\$=RIGHT\$(X\$,1) 'GET LAST CHAR
061E	16A6	IF LASTC\$<>" " AND LASTC\$<>"0" THEN PRIMDX\$="" 'REMOVE V OR S
064A	16A6	PRIMDX\$=LEFT\$(PRIMDX\$+X\$,5) 'GET 5 CHAR ONLY
065C	16A6	
065C	16A6	IF PRIMDX\$<>" " THEN 113
066A	16A6	N.ERR=N.ERR+1
0672	16A6	ED.MSG\$(N.ERR)="NO DIAGNOSIS CODED"
0686	16A6	

Offset	Data	Source Line
0686	16A6	113 REM
0687	16A6	DAT.OFFSET = 53
068E	16A8	PFX.OFFSET = 57
0695	16AA	PR1.OFFSET = 58
069C	16AC	PRT.OFFSET = 62
06A3	16AE	P2X.OFFSET = 63
06AA	1680	PR2.OFFSET = 64
0681	1682	P2T.OFFSET = 68
0688	1684	REA.OFFSET = 69
068F	1686	VCT.OFFSET = 70
06C6	1688	UPR.OFFSET = 71
06CD	168A	PRC.OFFSET = 76
06D4	168C	PC2.OFFSET = 102
06DB	168E	TOT.OFFSET = 75
06E2	16C0	POINTER = 0
06E9	16C2	
06E9	16C2	CMS=MID\$(DATE\$,1,2) 'CURRENT MONTH
06FB	16C6	FOR I910 = 1 TO 4 'INITIALIZE DATE ARRAY
0702	16C6	VDATE\$(I910)=" "
0716	16C8	NEXT I910
0725	16C8	
0725	16C8	REM *** REPEAT THE FOLLOWING CODE 4 TIMES ***
0725	16C8	FOR I910= 1 TO 4
072C	16C8	
072C	16C8	REM *** VISIT DATE ***
072C	16C8	YRS=MID\$(DATE\$,9,2) 'CURRENT YEAR
073E	16CC	XS=MID\$(TEXT\$,DAT.OFFSET+POINTER,4) 'MONTH AND DAY FROM FORM
0755	16CC	IF XS=" " THEN 199 'TIME TO QUIT
0763	16CC	IF LEFT\$(XS,2)<=CMS THEN 116 'OK, USE THIS YEAR
0778	16CC	YRS=RIGHT\$(STR\$(VAL(YRS)-1),2) 'USE LAST YEAR
0798	16CC	116 VDATE\$ = YRS + XS
07A6	16D0	'EDIT VISIT DATE
07A6	16D0	CK.5000\$=VDATE\$
07AF	16D4	GOSUB 5000 'DATE CHECK
07B4	16D4	CK.5010\$=VDATE\$
07BD	16D8	GOSUB 5010 'NUMERIC STRING CHECK
07C2	16D8	IF RT.5000 = 0 AND RT.5010 = 0 THEN GOTO 118
07E6	16DC	N.ERR=N.ERR+1
07EE	16DC	ED.MSG\$(N.ERR)="TODAYS DATE " + DATEERR\$(RT.5000)
0814	16DC	
0814	16DC	118 VDATE\$(I910)=VDATE\$
0828	16DC	
0828	16DC	REM *** PROVIDER 1 ID (PREFIX + NUM) ***
0828	16DC	120 PROV1.PFX\$ = MID\$(TEXT\$,PFX.OFFSET+POINTER,1)
083F	16E0	PROV1.NUM\$ = MID\$(TEXT\$,PR1.OFFSET+POINTER,4)
0856	16E4	PR1.COD\$(I910)=PROV1.PFX\$ + PROV1.NUM\$
086F	16E4	
086F	16E4	REM *** PRIMARY PROVIDER TIME ***
086F	16E4	122 X=VAL(MID\$(TEXT\$,PRT.OFFSET+POINTER,1))
088A	16E4	PR1.TIM\$(I910)=PROVIDER.TIM\$(X)
08A7	16E4	
08A7	16E4	REM *** PROVIDER 2 ID (PFX + NUM) ***
08A7	16E4	123 PROV2.PFX\$ = MID\$(TEXT\$,P2X.OFFSET+POINTER,1)
088E	16E8	PROV2.NUM\$ = MID\$(TEXT\$,PR2.OFFSET+POINTER,4)

Offset	Data	Source Line
0805	16EC	PR2.COD\$(I910)=PROV2.PFX\$ + PROV2.NUM\$
08EE	16EC	IF PR2.COD\$(I910)=" " THEN PB=1 ELSE PB=0 'PROV2 BLANK INDICATOR
0917	16EE	IX\$=STR\$(I910)
0924	16F2	IF PROV2.PFX\$=" " AND PROV2.NUM\$<>" " THEN 124
094A	16F2	IF PROV2.PFX\$<>" " AND PROV2.NUM\$=" " THEN 124
0970	16F2	GOTO 125
0974	16F2	
0974	16F2	124 N.ERR = N.ERR + 1
097C	16F2	ED.MSG\$(N.ERR)="PROV 2 CODE MISSING PREFIX OR NUMBER IN PART"+IX\$
0995	16F2	
0995	16F2	REM *** SECONDARY PROVIDER TIME ***
0995	16F2	125 X=VAL(MID\$(TEXT\$,P2T.OFFSET+POINTER,1))
09B0	16F2	PR2.TIM\$(I910)=PROVIDER.TIM\$(X)
09CD	16F2	REM IS THERE A TIME AND NO SEC PROV CODED?
09CD	16F2	IF PR2.TIM\$(I910)<>"000" AND PB=1 THEN 126
09FA	16F2	REM IS THERE NO TIME AND A SEC PROV CODED?
09FA	16F2	IF PR2.TIM\$(I910)="000" AND PB=0 THEN 127
0A27	16F2	GOTO 128
0A2B	16F2	
0A2B	16F2	126 N.ERR = N.ERR + 1
0A33	16F2	ED.MSG\$(N.ERR)="TIME CODED WITH NO SEC PROV CODED IN PART"+IX\$
0A4C	16F2	GO TO 128
0A50	16F2	
0A50	16F2	127 N.ERR = N.ERR + 1
0A58	16F2	ED.MSG\$(N.ERR)="NO PROV 2 TIME IN PART"+IX\$
0A71	16F2	
0A71	16F2	REM *** REASON FOR #2 CARE PROVIDER ***
0A71	16F2	128 SPROV.REAS\$(I910)=MID\$(TEXT\$,REA.OFFSET+POINTER,1)
0A94	16F2	REM IS THERE A REAS AND NO SEC PROV CODED?
0A94	16F2	IF SPROV.REAS\$(I910)<>" " AND PB=1 THEN 129
0AC1	16F2	REM IS THERE NO REAS AND A SEC PROV CODED?
0AC1	16F2	IF SPROV.REAS\$(I910)=" " AND PB=0 THEN 130
0AEE	16F2	GOTO 131
0AF2	16F2	
0AF2	16F2	129 N.ERR = N.ERR + 1
0AFA	16F2	ED.MSG\$(N.ERR)="REASON CODED WITH NO SEC PROV CODED IN PART"+IX\$
0B13	16F2	GO TO 131
0B17	16F2	
0B17	16F2	130 N.ERR = N.ERR + 1
0B1F	16F2	ED.MSG\$(N.ERR)="NO PROV 2 REASON IN PART"+IX\$
0B38	16F2	
0B38	16F2	
0B38	16F2	REM *** VISIT COUNT ***
0B38	16F2	131 VCNT\$="1" 'DEFAULT
0B41	16F6	X\$=MID\$(TEXT\$,VCT.OFFSET+POINTER,1)
0B58	16F6	IF X\$="1" THEN VCNT\$="0"
0B6F	16F6	IF X\$>"1" THEN VCNT\$=X\$
0B86	16F6	VIS.CNT\$(I910)=VCNT\$
0B9A	16F6	
0B9A	16F6	REM *** ADDITIONAL PROCEDURE
0B9A	16F6	132 ADDP\$=MID\$(TEXT\$,UPR.OFFSET+POINTER,5)
0BB1	16FA	
0BB1	16FA	REM *** PROCEDURE CODES FOR PROV 1 ***
0BB1	16FA	134 X\$ = MID\$(TEXT\$,PRC.OFFSET+POINTER,N.PROC)

Offset	Data	Source Line
0BC9	16FA	GOSUB 6000 'DECODE THE X\$ STRING
0BCE	16FA	IF TOT = 0 THEN GOTO 135
0BDD	16FC	FOR I910.A= 1 TO TOT
0BEA	16FE	PTR=VAL(HOLD\$(I910.A))
0C00	1702	GROUP1\$(I910,I910.A)=PROCD\$(PTR)
0C22	1702	NEXT I910.A
0C33	1702	
0C33	1702	REM ... ADD UNLISTED CODE IF THERE TO PROV 1 ...
0C33	1702	135 IF ADDP\$=" " THEN 136
0C41	1702	TOT=TOT+1
0C49	1702	GROUP1\$(I910,TOT)=ADDP\$
0C62	1702	
0C62	1702	136 TOT.PROC1(I910)=TOT
0C71	1702	
0C71	1702	REM *** PROCEDURE CODES FOR PROV 2 ***
0C71	1702	138 X\$ = MID\$(TEXT\$,PC2.OFFSET+POINTER,N.PROC)
0CB9	1702	GOSUB 6000 'DECODE THE X\$ STRING
0C8E	1702	IF TOT = 0 THEN GOTO 140
0C9D	1702	IF PB=1 THEN 141 'ERROR IF PROV 2 CODE BLANK
0CA8	1702	FOR I910.A= 1 TO TOT
0CB5	1704	PTR=VAL(HOLD\$(I910.A))
0CC8	1704	GROUP2\$(I910,I910.A)=PROCD\$(PTR)
0CED	1704	NEXT I910.A
0CFE	1704	
0CFE	1704	140 TOT.PROC2(I910)=TOT
0D0D	1704	GOTO 142
0D11	1704	
0D11	1704	141 N.ERR = N.ERR + 1
0D19	1704	ED.MSG\$(N.ERR)="PROC FOR SEC PROV BUT NO SEC PROV CODED IN PART"+IX\$
0D32	1704	
0D32	1704	142 POINTER = POINTER + TOT.OFFSET
0D3D	1704	NEXT I910
0D4F	1704	199 REM
0D50	1704	
0D50	1704	REM -----END OF MODULE RXXM920.M01-----
0D50	1704	
0D50	1704	IF N.ERR = 0 THEN GOTO 997
0D5F	1704	LPRINT "LITHO # ";LITHO\$;" ... ERRORS"
0D71	1704	FOR I997 = 1 TO N.ERR
0D7E	1706	LPRINT USING "### ";I997;
0D8A	1708	LPRINT "=> ";ED.MSG\$(I997)
0DA0	1708	NEXT I997
0DB1	1708	LN.COUNT = LN.COUNT + N.ERR + 1
0DB0	170A	CNTRLOPT = 6
0DC4	170A	GOSUB 9010 'REJECT THE FORM
0DC9	170A	GOTO 998 'BYPASS THE DISK WRITER....
0DCD	170A	
0DCD	170A	997 REM \$INCLUDE: 'RACH920.M02' REM INCLUDE THE BASE ENCOUNTER FORM DISK WRITER
0DCE	170A	REM *****
0DCE	170A	REM ***** AMBULATORY CARE INFORMATION SYSTEM 13 MAR 87 *****
0DCE	170A	REM ***** D R BOLLING *****
0DCE	170A	REM ***** MODULE NAME : RACH920.M02 *****
0DCE	170A	REM ***** SCANNER PROGRAM # : 920- PT REPEAT VISIT FORM *****

Offset Data Source Line IBM Personal Computer BASIC Compiler V1.00

```

00CE 170A REM ****
00CE 170A REM **** PURPOSE : CREATE AND WRITE THE DISK ****
00CE 170A REM **** RECORD FOR INPUT TO FOCUS ****
00CE 170A REM ****
00CE 170A REM **** PROGRAM ADDS PREFIX TO LITHO FOR EACH PATIENT ****
00CE 170A REM ****
00CE 170A REM **** RESERVED LINE NUMBERS 200-299 ****
00CE 170A REM ****
00CE 170A REM BUILD THE OUTPUT RECORD
00CE 170A REM ****
00CE 170A REM **** RECORD TYPE "1" - RECKEY PLUS TYPE 1 FIELDS ****
00CE 170A REM ****
00CE 170A REM RECOUNTS ="9201"+RECKEYS 'TRANSACTION IDENTIFIER
00CE 170A
00CE 170A REM ... TYPE 1 RECORD ...
00CE 170A
00CE 170A FOR IK = 1 TO 4
00D5 170A RECKEYS = CL1.CODS + VDATS(IK) +PR1.CODS(IK) + SSNS + FMEMP$
00E5 1710 IF VDATS(IK) = " " THEN 240
00E10 1710 RECOUNTS = "9201" + RECKEYS + RIGHTS(STR$(IK),1) + RIGHTS(LITHOS,7)
00E40 1714 TYPE1$ = "92" + VIS.CNTS(IK)+PR1.TIMS(IK)+PR2.CODS(IK)+PR2.TIMS(IK)
00E91 1718 TYPE1AS= SPROV.REAS(IK) + " " + REF.CODS + STRINGS(11," ") + PRIMDX$
00EC2 171C RECOUNTS = RECOUNTS + TYPE1$ + TYPE1AS
00ED5 171C GOSUB 280
00EDA 171C PRINT #1,RECOUNTS
00EE5 171C 240 NEXT IK
00EF7 171C
00EF7 171C REM ****
00EF7 171C REM **** RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE ****
00EF7 171C REM ****
00EF7 171C REM RECOUNTS ="9202"+RECKEYS 'TRANSACTION IDENTIFIER
00EF7 171C
00EF7 171C REM ... PROCEDURES OF FIRST PROVIDER ...
00EF7 171C
00EF7 171C FOR I910 = 1 TO 4
00EFE 171C FOR I910.A = 1 TO TOT.PROC1(I910)
00F12 171E RECKEYS = CL1.CODS + VDATS(I910) +PR1.CODS(I910) + SSNS + FMEMP$
00F42 171E IF VDATS(I910) = " " THEN 250
00F5A 171E RECOUNTS = "9202" + RECKEYS + RIGHTS(STR$(I910),1) + RIGHTS(LITHOS,7)
00F8A 171E RECOUNTS =RECOUNTS + "921" + GROUP1$(I910,I910.A)
00FAE 171E GOSUB 280
00FB3 171E PRINT #1,RECOUNTS
00FBE 171E NEXT I910.A
00FD2 171E 250 NEXT I910
00FE4 171E
00FE4 171E REM ... PROCEDURES OF SECOND PROVIDER ...
00FE4 171E
00FE4 171E FOR I910 = 1 TO 4
00FEB 171E FOR I910.A = 1 TO TOT.PROC2(I910)
00FFF 1720 RECKEYS = CL1.CODS + VDATS(I910) +PR1.CODS(I910) + SSNS + FMEMP$
102F 1720 IF VDATS(I910) = " " THEN 260
1047 1720 RECOUNTS = "9202" + RECKEYS + RIGHTS(STR$(I910),1) + RIGHTS(LITHOS,7)

```

Offset	Data	Source Line
1077	1720	RECOUT\$ = RECOUT\$ + "922" + GROUP2\$(1910,1910.A)
1098	1720	GOSUB 280
10A0	1720	PRINT #1,RECOUT\$
10A8	1720	NEXT 1910.A
10BF	1720	260 NEXT 1910
10D1	1720	
10D1	1720	GOTO 299
10D5	1720	
10D5	1720	REM *****
10D5	1720	REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
10D5	1720	REM *****
10D5	1720	280 PAD=MAXLENGTH - LEN(RECOUT\$) 'FIND OUT HOW SHORT THE RECORD IS
10E3	1722	RECOUT\$ = RECOUT\$ + STRING\$(PAD,PAD\$) 'PAD THE RECORD WITH FILL CHAR
10F6	1722	RETURN
10F9	1722	
10F9	1722	
10F9	1722	299 REM
10FA	1722	
10FA	1722	REM -----END OF MODULE RXXM920.MO2-----
10FA	1722	
10FA	1722	998 REM CONTINUE
10FB	1722	
10FB	1722	999 READTYPE = 2
1102	1722	IF LN.COUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
1112	1722	GOTO 10
1116	1722	
1116	1722	REM END OF SCAN/DECODE/WRITE LOOP =====
1116	1722	
1116	1722	1000 REM \$INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
1117	1722	REM *****
1117	1722	REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
1117	1722	REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
1117	1722	REM *****
1117	1722	REM PATIENT OMR INPUT PROGRAM *
1117	1722	REM *
1117	1722	REM This program verifies user is logged on properly. If there is no *
1117	1722	REM valid user logged on at the time of execution, this subroutine will*
1117	1722	REM chain to the logon program RACP05, otherwise a return is issued. *
1117	1722	REM *****
1117	1722	REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
1117	1722	REM *****
1117	1722	1001 OPEN "I",1,"RACLOG.DAT"
1129	1722	IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
1137	1722	INPUT #1,USER\$(1),DT\$,TMS\$,PID\$
1158	172E	IF USER\$(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
1166	172E	IF USER\$(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
1174	172E	CLOSE 1
1178	172E	SCREEN 0,1,0,0
1191	172E	COLOR FORE,BACK,BORD
11A7	172E	CLS
11AB	172E	RETURN
11AE	172E	
11AE	172E	1002 CLOSE
11B2	172E	CHAIN "RACP05"

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

1189 172E '=====END OF LOGON VERIFY SUBROUTINE 1000=====
1189 172E
1189 172E 2000 REM $INCLUDE: 'RACS2000.SUB' INCLUDE THE REPLY/DELAY SUB
118A 172E REM *****
118A 172E REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
118A 172E REM **** SKIP COLE ****
118A 172E REM **** SUBROUTINE NAME : RACS2000.SUB ****
118A 172E REM **** SCANNER PROGRAM # : ALL ****
118A 172E REM **** FUNCTION : THIS SUBROUTINE MODULE ****
118A 172E REM **** SERVERS AS A WAIT AND REPLY ****
118A 172E REM **** ENTRY MODULE ****
118A 172E REM **** INPUT : SINGLE KEYBOARD ENTRY ****
118A 172E REM **** ****
118A 172E REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
118A 172E REM **** ****
118A 172E REM **** RESERVED LINE ****
118A 172E REM **** NUMBERS : 2001-2010 ****
118A 172E REM *****
118A 172E 2001 REM REPLY FUNCTION
1188 172E 2002 REPLY$=INKEY$ : IF REPLY$="" THEN 2002
11CF 1732 REPLY=ASC(REPLY$)
11D9 1734 IF REPLY > 90 THEN REPLY$=CHR$(REPLY XOR 32) 'CONVERT TO CAPS
11F4 1734 IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$=""
1220 1734 RETURN
1223 1734
1223 1734 5000 REM $INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB
1224 1734 REM *****
1224 1734 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
1224 1734 REM **** SKIP COLE ****
1224 1734 REM **** SUBROUTINE NAME : RXXS5000.SUB ****
1224 1734 REM **** SCANNER PROGRAM # : ALL ****
1224 1734 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1224 1734 REM **** PERFORMS A DATE EDIT ****
1224 1734 REM **** ****
1224 1734 REM **** INPUT : DATE TO BE CHECKED MUST BE ****
1224 1734 REM **** IN THE VARIABLE NAMED ****
1224 1734 REM **** 'CK.5000$' ****
1224 1734 REM **** IN THE FORMAT "YYMMDD" ****
1224 1734 REM **** ****
1224 1734 REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
1224 1734 REM **** VARIABLE. IF THIS VARIABLE ****
1224 1734 REM **** CONTAINS ANY NUMBER OTHER ****
1224 1734 REM **** THAN 0, AN ERROR WAS FOUND ****
1224 1734 REM **** IN THE DATE. ****
1224 1734 REM **** ****
1224 1734 REM **** RESERVED LINE ****
1224 1734 REM **** NUMBERS : 5001-5009 ****
1224 1734 REM *****
1224 1734 RT.5000 = 0
1228 1734 CKYEAR = VAL(LEFT$(CK.5000$,2)) 'YEAR NUMERIC VALUE
123E 1736 CKMONTH = VAL(MID$(CK.5000$,3,2)) 'MONTH NUMERIC VALUE
1254 1738 CKDAY = VAL(RIGHT$(CK.5000$,2)) 'DAY NUMERIC VALUE
1267 173A
1267 173A IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009

```

```

Offset Data Source Line
127D 173A IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
1293 173A IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
12A9 173A IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
12BF 173A
12BF 173A REM LEAP YEAR CHECK
12BF 173A MOLENGTH(2) = 28
12C6 173A IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY
12D5 173A IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR
12EA 173A MOLENGTH(2) = 29
12F1 173A
12F1 173A 5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
1310 173A
1310 173A 5009 RETURN
1313 173A
1313 173A REM -----END OF SUBROUTINE 5000 -----
1313 173A
1313 173A 5010 REM $INCLUDE: 'RACS5010.SUB' INCLUDE THE NUMERIC STRING EDITOR
1314 173A REM *****
1314 173A REM **** AMBULATORY CARE DATA BASE 1 MAY 85 ****
1314 173A REM **** SKIP COLE ****
1314 173A REM **** SUBROUTINE NAME : RXXS5010.SUB ****
1314 173A REM **** SCANNER PROGRAM # : ALL ****
1314 173A REM **** FUNCTION : THIS SUBROUTINE MODULE ****
1314 173A REM **** PERFORMS A NUMERIC STRING ****
1314 173A REM **** EDIT. ****
1314 173A REM ****
1314 173A REM **** INPUT : STRING TO BE EDITED IS IN ****
1314 173A REM **** THE VARIABLE NAMED ****
1314 173A REM **** 'CK.5010$' ****
1314 173A REM ****
1314 173A REM **** OUTPUT : 'RT.5010' IS THE RETURN CODE ****
1314 173A REM **** VARIABLE. IF THIS VARIABLE ****
1314 173A REM **** CONTAINS ANY NUMBER OTHER ****
1314 173A REM **** THAN 0, AN ERROR WAS FOUND ****
1314 173A REM **** IN THE STRING. ****
1314 173A REM ****
1314 173A REM **** RESERVED LINE ****
1314 173A REM **** NUMBERS : 5011-5019 ****
1314 173A REM *****
1314 173A RT.5010 = 0
1318 173A
1318 173A FOR I5010 = 1 TO LEN(CK.5010$)
1328 173C J5010= ASC(MID$(CK.5010$,I5010,1))
133F 1740 IF J5010 < 48 OR J5010 > 57 THEN RT.5010 = RT.5010 + 1
1367 1740 NEXT I5010
1378 1740
1378 1740 RETURN
1378 1740 REM ----- END OF SUBROUTINE 5010 -----
1378 1740 5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTPATIENT UCA CHECK SUB
137C 1740
137C 1740 6000 REM $INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE ROUTINE
137D 1740 REM *****
137D 1740 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
137D 1740 REM **** SKIP COLE ****

```

RACP920

PT REPEAT VISIT FORM

PAGE 15

04-28-87

09:28:25

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

137D 1740 REM **** SUBROUTINE NAME      :  RXXS6000.SUB      ****
137D 1740 REM **** SCANNER PROGRAM #    :  ALL              ****
137D 1740 REM **** FUNCTION              :  THIS SUBROUTINE MODULE ****
137D 1740 REM ****                      :  PERFORMS INSTRING SEARCH ****
137D 1740 REM ****                      :                      ****
137D 1740 REM **** INPUT                  :  STRING TO BE SEARCHED MUST ****
137D 1740 REM ****                      :  BE IN THE VARIABLE NAMED : ****
137D 1740 REM ****                      :  'XS'              ****
137D 1740 REM ****                      :                      ****
137D 1740 REM **** OUTPUT                  :  'TOT' = TOTAL NUMBER OF ****
137D 1740 REM ****                      :  HITS IN THE DESTING ****
137D 1740 REM ****                      :  'HOLD$( )' IS THE ARRAY ****
137D 1740 REM ****                      :  CONTAINING THE NUMERIC ****
137D 1740 REM ****                      :  VALUE OF THE HIT POSITIONS ****
137D 1740 REM ****                      :                      ****
137D 1740 REM ****                      :  *****
137D 1740 REM **** RESERVED LINE          :                      ****
137D 1740 REM **** NUMBERS                  :  6001-6009      ****
137D 1740 REM ****                      :  *****
137D 1740 6001      PTR = INSTR(X$, "1")
1388 1740          TOT = 0
1392 1740          WHILE PTR > 0
139D 1740              TOT=TOT+1
13A5 1740              HOLD$(TOT) = RIGHT$(STR$(PTR),2)
13C7 1740              PTR=PTR+1
13CF 1740              PTR = INSTR(PTR,X$, "1")
13E1 1740          WEND
13E5 1740          RETURN
13E8 1740
13E8 1740 REM -----END OF SUBROUTINE RXXS6000.SUB-----
13E8 1740
13E8 1740 7000 REM $INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
13E9 1740 REM *****
13E9 1740 REM **** AMBULATORY CARE DATA BASE      13 APR 85 ****
13E9 1740 REM ****                                SKIP COLE ****
13E9 1740 REM **** SUBROUTINE NAME      :  RACS7000.SUB      ****
13E9 1740 REM **** SCANNER PROGRAM #    :  ALL              ****
13E9 1740 REM **** FUNCTION              :  THIS SUBROUTINE MODULE ****
13E9 1740 REM ****                      :  PRINTS THE STANDARD SCREEN ****
13E9 1740 REM ****                      :  HEADING. ****
13E9 1740 REM **** INPUT                  :  COMMON VARIABLE USERS(2) ****
13E9 1740 REM ****                      :  SYSTEM DATE ****
13E9 1740 REM ****                      :                      ****
13E9 1740 REM **** OUTPUT                  :  SCREEN HEADING ****
13E9 1740 REM ****                      :                      ****
13E9 1740 REM **** RESERVED LINE          :                      ****
13E9 1740 REM **** NUMBERS                  :  7001-7010      ****
13E9 1740 REM ****                      :  *****
13E9 1740
13E9 1740 7001 LOCATE 1,1
13F3 1740      PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
13FB 1740      LOCATE 1,65
1408 1740      PRINT DATES;

```

Offset	Data	Source Line
1410	1740	LOCATE 2,1
141D	1740	PRINT "USER : ";USERS(1)
142A	1740	RETURN
142D	1740	7100 REM \$INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
142E	1740	REM *****
142E	1740	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
142E	1740	REM **** SKIP COLE ****
142E	1740	REM **** SUBROUTINE NAME : RXXS7100.SUB ****
142E	1740	REM **** SCANNER PROGRAM # : ALL ****
142E	1740	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
142E	1740	REM **** PRINTS THE STANDARD HEADING ****
142E	1740	REM **** ON THE PRINTER. ****
142E	1740	REM **** INPUT : DATE,PAGE,PGMIDS,PGMTITL\$ ****
142E	1740	REM **** ****
142E	1740	REM **** OUTPUT : PRINTER HEADING, LN.COUNT ****
142E	1740	REM **** ****
142E	1740	REM **** RESERVED LINE ****
142E	1740	REM **** NUMBERS : 7101-7110 ****
142E	1740	REM *****
142E	1740	
142E	1740	7101 IF PAGE > 0 THEN LPRINT CHR\$(12);
1444	1740	LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITL\$;
1451	1740	LPRINT TAB(70);DATES
1464	1740	PAGE=PAGE+1
146C	1740	LPRINT "PROGRAM ";PGMIDS;TAB(70);"PAGE";
1489	1740	LPRINT USING "####";PAGE
1495	1740	LPRINT
149D	1740	LN.COUNT=3
14A4	1740	RETURN
14A7	1740	
14A7	1740	8000 REM \$INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
14A8	1740	REM *****
14A8	1740	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ***
14A8	1740	REM **** SKIP COLE ***
14A8	1740	REM **** SUBROUTINE NAME : RXXS8000.SUB ***
14A8	1740	REM **** SCANNER PROGRAM # : ALL ***
14A8	1740	REM **** FUNCTION : THIS SUBROUTINE MODULE ***
14A8	1740	REM **** IS A GROUPING THAT PERFORMS ***
14A8	1740	REM **** VARIOUS DECODING FUNCTIONS ***
14A8	1740	REM **** ON THE SCANNER DATA ***
14A8	1740	REM **** ****
14A8	1740	REM **** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
14A8	1740	REM **** 8050 - CHECK FOR END OF JOB ***
14A8	1740	REM **** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
14A8	1740	REM **** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
14A8	1740	REM **** (RETURNED IN TEXT\$ STRING VARIABLE) ***
14A8	1740	REM **** ****
14A8	1740	REM **** INPUT : SHEET RECORD, RECORD LENGTH ***
14A8	1740	REM **** ****
14A8	1740	REM **** OUTPUT : 'TEXT\$' TRING VARIABLE ***
14A8	1740	REM **** ****
14A8	1740	REM **** RESERVED LINE ***
14A8	1740	REM **** NUMBERS : 8001-8500 ***
14A8	1740	REM *****

Offset	Data	Source Line
14A8	1740	
14A8	1740	'DECODE THE HEADER ONLY
14A8	1740	8001 POINTER = 0
14AF	1740	RECORDPTR = VARPTR(SHEETREC(0))
14B6	1742	FOR J8000 = 1 TO 21
14B0	1742	8002 TEXT\$= TEXT\$+CHR\$(PEEK(RECORDPTR + POINTER))
14DB	1742	POINTER=POINTER+1
14E3	1742	NEXT J8000
14F2	1744	PROGRAM\$= LEFT\$(TEXT\$,3)
1501	1748	BATCH\$= MID\$(TEXT\$,4,3)
1513	174C	SERIAL\$= MID\$(TEXT\$,7,4)
1525	1750	RUNID\$= MID\$(TEXT\$,11,1)
1537	1754	FORM\$= MID\$(TEXT\$,12,2)
1549	1758	POCKET\$= MID\$(TEXT\$,14,1)
1558	175C	SCANERR1\$=MID\$(TEXT\$,16,2)
1560	1760	SCANERR2\$=MID\$(TEXT\$,18,2)
157F	1764	SCANERR3\$=MID\$(TEXT\$,20,2)
1591	1768	GOTO 8500
1595	1768	
1595	1768	8050 REM CHECK FOR END OF JOB/END OF BATCH
1596	1768	IF PROGRAM\$ = PGMI\$ THEN GOTO 8500
15A8	1768	LPRINT STRINGS(80,"**")
15B6	1768	LPRINT
15BE	1768	LPRINT "RECORDS PROCESSED ... ";SERIAL\$
15CB	1768	LPRINT "STARTED AT ";BTIM\$
15D8	1768	LPRINT "ENDED AT ";TIM\$
15E5	1768	LPRINT CHR\$(12)
15F0	1768	GOTO 30000
15F4	1768	
15F4	1768	8070 REM CHECK FOR SCANNER ERRORS
15F5	1768	IF POCKET\$ = " " GOTO 8500
1607	1768	LPRINT LITHOS;
160F	1768	LPRINT " ... SCANNER ERRORS : ";
1617	1768	LPRINT SCANERR1\$;" / ";
1624	1768	LPRINT SCANERR2\$;" / ";
1631	1768	LPRINT SCANERR3\$
1639	1768	LN=LN+1
1641	176A	GOTO 999
1645	176A	
1645	176A	8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
1646	176A	LOCATE 5,1:PRINT "PROGRAM ";PROGRAM\$;
1660	176A	
1660	176A	PRINT " BATCH ";BATCH\$;
1660	176A	PRINT " RUN ";RUNID\$;
167A	176A	PRINT " FORM ";FORM\$;
1687	176A	PRINT " POCKET ";POCKET\$
1694	176A	GOTO 8500
1698	176A	
1698	176A	8200 REM DECODE THE RESPONSE POSITIONS
1699	176A	POINTER = 21
16A0	176A	RECORDPTR = VARPTR(SHEETREC(0))
16A7	176A	FOR J8000 = 22 TO RECORDLENGTH
1684	176C	8202 TEXT\$ = TEXT\$+CHR\$(PEEK(RECORDPTR + POINTER))
16D2	176C	POINTER=POINTER+1

Offset	Data	Source Line
16DA	176C	NEXT J8000
16EB	176C	
16EB	176C	8500 RETURN
16EE	176C	
16EE	176C	REM ----- END OF RXXS8000.SUB -----
16EE	176C	
16EE	176C	9000 REM \$INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
16EF	176C	REM *****
16EF	176C	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
16EF	176C	REM **** SKIP COLE ****
16EF	176C	REM **** PROGRAM NAME : RACS9000.SUB ****
16EF	176C	REM **** SCANNER PROGRAM # : ALL ****
16EF	176C	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
16EF	176C	REM **** CONTROLS THE SCANNER I/O ****
16EF	176C	REM ****
16EF	176C	REM **** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS ****
16EF	176C	REM **** COMMUNICATIONS MANUAL AND THE ****
16EF	176C	REM **** PRE-RELEASED SOFTWARE GUIDE ****
16EF	176C	REM ****
16EF	176C	REM *****
16EF	176C	REM **** RESERVED LINE ****
16EF	176C	REM **** NUMBERS : 9001-9100 ****
16EF	176C	REM *****
16EF	176C	
16EF	176C	REM *****
16EF	176C	REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
16EF	176C	REM **** ARGUMENTS: PRESET ... SEE BELOW ****
16EF	176C	REM *****
16EF	176C	9001 REM
16F0	176C	PROTOCOL(0) = 9600 'BAUD RATE
16F7	176C	PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
16FE	176C	PROTOCOL(2) = 8 'DATA BITS
1705	176C	PROTOCOL(3) = 1 'STOP BITS
170C	176C	PROTOCOL(4) = 2 'RS-232 PORT
1713	176C	PROTOCOL(5) = 0 'WRITE TIME-OUT
171A	176C	PROTOCOL(6) = 0 'READ TIME-OUT
1721	176C	
1721	176C	ERRSTAT\$ = SPACE\$(60)
172D	176C	ARGPTR = VARPTR(PROTOCOL(0))
1734	176E	CALL SETUP (ARGPTR,ERRSTAT\$)
1745	176E	ERRMSG\$=""
174E	176E	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="SETUP ERROR "+ERRSTAT\$
176A	176E	GOTO 9100
176E	176E	
176E	176E	REM *****
176E	176E	REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
176E	176E	REM **** ARGUMENTS: CNTRLOPT ****
176E	176E	REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
176E	176E	REM **** CNTRLOPT = 2 = STOP SCANNER (S0) ****
176E	176E	REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
176E	176E	REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
176E	176E	REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
176E	176E	REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

176E 176E REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) *****
176E 176E REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) *****
176E 176E REM *****
176E 176E 9010 REM
176F 176E ERRSTAT$ = SPACES(60)
177B 176E CALL CNTRL (CNTRLOPT,ERRSTAT$)
178C 176E ERRMSG$=""
1795 176E IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="CONTROL ERROR "+ERRSTAT$
17B1 176E GOTO 9100
17B5 176E
17B5 176E REM *****
17B5 176E REM * ** SUBROUTINE 9020 - SCAN SHEET CALL *****
17B5 176E REM *****
17B5 176E REM **** ARGUMENTS: READTYPE *****
17B5 176E REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER *****
17B5 176E REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT *****
17B5 176E REM *****
17B5 176E REM **** ARGUMENTS: RECORDLENGTH *****
17B5 176E REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE *****
17B5 176E REM **** TRANSMITTED *****
17B5 176E REM *****
17B5 176E 9020 REM
17B6 176E ERRSTAT$ = SPACES(60)
17C2 176E RECORDPTR = VARPTR(SHEETREC(0))
17C9 176E CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT$)
17E2 176E ERRMSG$=""
17EB 176E IF MID$(ERRSTAT$,14,3) = "415" THEN ERRMSG$="ESC"
180C 176E GOTO 9100
1810 176E
1810 176E REM *****
1810 176E REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL *****
1810 176E REM *****
1810 176E REM **** ARGUMENTS: PRINTPOS *****
1810 176E REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION *****
1810 176E REM **** VALUES = 0 THRU 90 *****
1810 176E REM *****
1810 176E REM **** ARGUMENTS: PSTRINGS *****
1810 176E REM **** TEXT TO BE PRINTED ON THE FORM *****
1810 176E REM *****
1810 176E REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN *****
1810 176E REM **** HEADER SHEET IS MARKED 'PRINTER ON' *****
1810 176E REM *****
1810 176E 9030 REM
1811 176E ERRSTAT$ = SPACES(60)
1810 176E RECORDPTR = VARPTR(SHEETREC(0))
1824 176E CALL TPRINT(PRINTPOS,PSTRINGS,ERRSTAT$)
1839 1774 IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="PRINT ERROR "+ERRSTAT$
1855 1774 GOTO 9100
1859 1774
1859 1774 9100 RETURN
185C 1774 REM -----END OF SUBROUTINE RACS9000.SUB -----
185C 1774
185C 1774 REM END OF SUBROUTINES =====
185C 1774

```

RACP920
PT REPEAT VISIT FORM

PAGE 20
04-28-87
09:28:25

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```
185C 1774 25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
185D 1774 LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1865 1774 LPRINT "ERASING FILE ";DATFIL$
1872 1774 BEEP
1876 1774 CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1882 1774 CLOSE
1886 1774 OPEN DATFIL$ FOR OUTPUT AS #1
1898 1774 PRINT #1,STRING$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
18AA 1774 CLOSE
18AE 1774
18AE 1774 30000 REM
18AF 1774 CLOSE
1883 1774 CHAIN "RACP10"
188A 1774 END
188E 1774
18C1 1774
```

22151 Bytes Available
15807 Bytes Free

0 Warning Error(s)
0 Severe Error(s)

PAGE 1

07-06-87

09:25:22

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```
001A 0002 REM $LINESIZE: 132
001A 0002 REM $PAGESIZE: 66
001A 0002 REM $TITLE: 'RACP930 '
001A 0002 REM $SUBTITLE: 'REPEAT ENCOUNTER DESTRING/DECODE PROGRAM'
001A 0002 REM $PAGE
```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

001A 0002 REM +-----+
001A 0002 REM | NAME: RACP930          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | Date: 11 MAR 87          REPEAT ENCOUNTER FORM          |
001A 0002 REM | D R BOLLING                                     |
001A 0002 REM +-----+
001A 0002 REM          REPEAT ENCOUNTER FORM INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the REPEAT ENCOUNTER form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RACP05, otherwise,
001A 0002 REM the program chains to program RACP10 on exit.
001A 0002 REM
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 REM *****
001A 0002 REM * NAME: RACDIM.MOD          DIMENSION DEFINITIONS *
001A 0002 REM * Date: 28 Feb 84          Written by: Floyd Cole *
001A 0002 REM *****
001A 0002 REM ' Dimensioned variables are defined in this file.
001A 0002 REM ' It is an included file so it cannot be run in a stand-alone,
001A 0002 REM ' mode.
001A 0002 REM '
001A 0002 REM ' This program segment may be modified, but all files containing
001A 0002 REM ' an include for this segment must be re-compiled in order to
001A 0002 REM ' affect the changes made here.
001A 0002 REM ' ***** START OF DIMENSION DEFINITION *****
001A 0002 REM
001A 0002 REM DEFINT A-Z
001A 0002 REM DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002 REM
001A 0002 REM ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002 REM
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002 REM
001A 0002 REM DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 REM DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 REM DIM ED.MSG$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002 REM DIM CLINIC1.PFX$(5) '(PREFIX -B D F G S- FOR CLINIC #1)
001A 0002 REM DIM DIAG.PFX$(2) '(DIAGNOSIS PREFIX TABLE)
001A 0002 REM DIM REFER.PFX$(6) '(REFERRAL PREFIX TABLE -A B C D F S-)
001A 0002 REM DIM PROV.TIMES(21) 'TIME SPENT
001A 0002 REM DIM CLIBUB$(11) 'SINGLE BUBBLE CLINIC CODE TABLE
001A 0002 REM DIM PVDC$(5) '(HOLD AREA FOR VISIT DATES AND PROV CODES)

```

```

Offset  Data  Source Line
001A 0002
001A 0032 REM $INCLUDE: 'RACCMN.MOD'      REM Include the COMMON AREA DEFINITION
001A 0002 *****
001A 0002 '* NAME: RACCMN.MOD                COMMON AREA DEFINITION *
001A 0002 '* Date: 28 Feb 84                Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A 0002 ' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re*compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 '
001A 0002 ' *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002 COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL$ 'BASIC SCREEN COLORS
001A 0002 COMMON HEADERS$ '21 CHARACTER SCANNER HEADER INFO
001A 0002 COMMON TEXT$ ' ' AINING CHARACTERS FROM SCANNER
001A 0002 COMMON PGMID$ 'PROGRAM OR FORM ID
001A 0002 COMMON MOLENGTH( ) 'DAYS IN THE MONTH
001A 0002 COMMON USER$( )
001A 0002 ' *****END OF COMMON DEFINITION*****
001A 0002
001A 0002 REM $INCLUDE: 'RACDEF.MOD'      REM Include the DEFAULT DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACP01.DEF              DEFAULT DEFINITIONS *
001A 0002 '* Date: 28 Feb 84              Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' Variables used in common that have a default value on start*up
001A 0002 ' will be held in this file. It is an included file so it cannot
001A 0002 ' be run in a stand*alone mode. In normal operation, this file
001A 0002 ' should be 'included' in the main program only (RACP10.BAS).
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re*compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 '
001A 0002 ' *****START OF DEFAULT DEFINITION*****
001A 0002
001A 0002 FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
0046 0F2C BACK = 1 'Background Color = Light Blue
004D 0F2C BORD = 4 'BORDER = RED
0054 0F2E HIDE = 4 'ALTERNATE COLOR = RED
0058 0F2E EFORE = 14 'ERROR FOREGROUND DISPLAY
0062 0F2E EBACK = 0 'ERROR BACKGROUND DISPLAY
0069 0F2E BELL$ = CHR$(7) 'Sound the bell
0075 0F2E
0075 0F2E MOLENGTH(1) = 31 'JAN
007C 0F2E MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
0083 0F2E MOLENGTH(3) = 31 'MAR
008A 0F2E MOLENGTH(4) = 30 'APR
0091 0F2E MOLENGTH(5) = 31 'MAY
0098 0F2E MOLENGTH(6) = 30 'JUN

```

Offset	Data	Source Line
009F	0F2E	MOLENGTH(7) = 31 'JUL
00A6	0F2E	MOLENGTH(8) = 31 'AUG
00AD	0F2E	MOLENGTH(9) = 30 'SEP
00B4	0F2E	MOLENGTH(10) = 31 'OCT
00BB	0F2E	MOLENGTH(11) = 30 'NOV
00C2	0F2E	MOLENGTH(12) = 31 'DEC
00C9	0F2E	
00C9	0F2E	DATEERR\$(0) = " "
00D2	0F2E	DATEERR\$(1) = "INVALID MONTH"
00DB	0F2E	DATEERR\$(2) = "INVALID DAY "
00E4	0F2E	DATEERR\$(3) = "DAY TOO LARGE FOR MONTH CODED"
00ED	0F2E	
00ED	0F2E	MAXLENGTH = 80 'MAXIMUM LENGTH OF OUTPUT RECORD
00F4	0F30	PAD\$ = "." 'PAD CHARACTER FOR SHORT RECORDS
00FD	0F34	
00FD	0F34	*****END OF DEFAULT DEFINITION*****
00FD	0F34	
00FD	0F34	
00FD	0F34	KEY OFF
0103	0F34	
0103	0F34	REM *****
0103	0F34	REM THE FOLLOWING VARIABLES ARE UNIQUE TO EACH PROGRAM AND MUST
0103	0F34	REM BE CHANGED.
0103	0F34	REM *****
0103	0F34	PGMTITL\$ = "REPEAT ENCOUNTER FORM"
010C	0F38	
010C	0F38	PGMID\$ = "930" 'VALUE RECEIVED FROM THE SCANNER
0115	0F38	'IN HEADER VARIABLE 'PROGRAMS'
0115	0F38	
0115	0F38	DATFIL\$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
011E	0F3C	
011E	0F3C	REM LENGTH OF STRING RECEIVED FROM THE OMR....
011E	0F3C	HEADER = 21
0125	0F3E	RESPONSE= 97
012C	0F40	RECORDLENGTH = HEADER + RESPONSE
0137	0F42	
0137	0F42	N.PROC = 1 'NUMBER OF PROCEDURES FOR THIS FORM
013E	0F44	
013E	0F44	REM *****
013E	0F44	
013E	0F44	BTIME\$=TIME\$ 'SCAN START TIME
0147	0F48	
0147	0F48	
0147	0F48	
0147	0F48	REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0147	0F48	REM CLINIC #1
0147	0F48	CLINIC1.PFX\$(0)="?"
0150	0F48	CLINIC1.PFX\$(1)="B"
0159	0F48	CLINIC1.PFX\$(2)="D"
0162	0F48	CLINIC1.PFX\$(3)="F"
0163	0F48	CLINIC1.PFX\$(4)="G"
0174	0F48	CLINIC1.PFX\$(5)="S"
017D	0F48	
017D	0F48	REM *** DIAGNOSIS PREFIX TABLE ***

Offset	Data	Source Line
017D	0F48	
017D	0F48	DIAG.PFX\$(0)="?"
0186	0F48	DIAG.PFX\$(1)="V"
018F	0F48	DIAG.PFX\$(2)="S"
0198	0F48	
0198	0F48	REM *** REFERRAL PREFIX TABLE ***
0198	0F48	
0198	0F48	REFER.PFX\$(0)=" "
01A1	0F48	REFER.PFX\$(1)="A"
01AA	0F48	REFER.PFX\$(2)="B"
01B3	0F48	REFER.PFX\$(3)="C"
01BC	0F48	REFER.PFX\$(4)="D"
01C5	0F48	REFER.PFX\$(5)="F"
01CE	0F48	REFER.PFX\$(6)="S"
01D7	0F48	
01D7	0F48	REM *** TIME SPENT TABLE ***
01D7	0F48	PROV.TIMES\$(00)="000" 'NO TIME
01E0	0F48	PROV.TIMES\$(01)="005" ' 5 MINUTES
01E9	0F48	PROV.TIMES\$(02)="010" '10 MINUTES
01F2	0F48	PROV.TIMES\$(03)="015" '15 MINUTES
01FB	0F48	PROV.TIMES\$(04)="020" '20 MINUTES
0204	0F48	PROV.TIMES\$(05)="030" '30 MINUTES
020D	0F48	PROV.TIMES\$(06)="045" '45 MINUTES
0216	0F48	PROV.TIMES\$(07)="060" ' 1 HOUR
021F	0F48	PROV.TIMES\$(08)="090" ' 1.5 HOURS
0228	0F48	PROV.TIMES\$(09)="120" ' 2 HOURS
0231	0F48	PROV.TIMES\$(10)="150" ' 2.5 HOURS
023A	0F48	PROV.TIMES\$(11)="180" ' 3 HOURS
0243	0F48	PROV.TIMES\$(12)="210" ' 3.5 HOURS
024C	0F48	PROV.TIMES\$(13)="240" ' 4 HOURS
0255	0F48	PROV.TIMES\$(14)="270" ' 4.5 HOURS
025E	0F48	PROV.TIMES\$(15)="300" ' 5 HOURS
0267	0F48	PROV.TIMES\$(16)="330" ' 5.5 HOURS
0270	0F48	PROV.TIMES\$(17)="360" ' 6 HOURS
0279	0F48	PROV.TIMES\$(18)="390" ' 6.5 HOURS
0282	0F48	PROV.TIMES\$(19)="420" ' 7 HOURS
0288	0F48	PROV.TIMES\$(20)="450" ' 7.5 HOURS
0294	0F48	PROV.TIMES\$(21)="480" ' 8 HOURS
029D	0F48	
029D	0F48	REM *** SINGLE BUBBLE CLINIC CODE TABLE ***
029D	0F48	CLIBUB\$(1)="BABA" : CLIBUB\$(7) ="BFDA"
02AF	0F48	CLIBUB\$(2)="BAJA" : CLIBUB\$(8) ="BHBA"
02C1	0F48	CLIBUB\$(3)="BAKA" : CLIBUB\$(9) ="DCBA"
02D3	0F48	CLIBUB\$(4)="BALA" : CLIBUB\$(10)="DHBA"
02E5	0F48	CLIBUB\$(5)="BAPA" : CLIBUB\$(11)="DHDA"
02F7	0F48	CLIBUB\$(6)="BFBA"
0300	0F48	
0300	0F48	REM INCLUDE: 'UCABAMC.OPT' INCLUDE OUTPATIENT UCA TABLE
0300	0F48	
0300	0F48	REM INCLUDE: 'UCABAMC.IPT' INCLUDE INPATIENT UCA TABLE
0300	0F48	
0300	0F48	REM \$PAGE

Offset	Data	Source Line
0300	0F48	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
0305	0F48	CLS
0309	0F48	GOSUB 7000 'PRINT SCREEN HEADING
030E	0F48	
030E	0F48	REM *****
030E	0F48	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
030E	0F48	REM *****
030E	0F48	REM
030E	0F48	OPEN DATFIL\$ FOR APPEND AS #1
0320	0F48	
0320	0F48	REM *****
0320	0F48	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
0320	0F48	REM *****
0320	0F48	LPRINT CHR\$(15);
0328	0F48	WIDTH "LPT1:",160
0335	0F48	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
0341	0F4A	COLOR 14
0348	0F4A	LOCATE 11,26 : PRINT "REPEAT VISIT FORM "
0350	0F4A	COLOR FORE,BACK,BORD
0373	0F4A	
0373	0F4A	REM *****
0373	0F4A	REM **** COMMUNICATIONS SETUP ****
0373	0F4A	REM *****
0373	0F4A	REM PROTOCOL
0373	0F4A	GOSUB 9001
0378	0F4A	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0392	0F4E	
0392	0F4E	REM START SCANNER (S1)
0392	0F4E	CNTRLOPT =1 :GOSUB 9010
039E	0F50	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0388	0F50	
0388	0F50	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
03CD	0F50	READYTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
03D4	0F52	
03D4	0F52	REM *****
03D4	0F52	REM **** SET SCAN SHEET CALL ****
03D4	0F52	REM *****
03D4	0F52	REM
03D4	0F52	
03D4	0F52	10 REM - RETURN POINT TO READ NEXT SHEET
03D5	0F52	
03D5	0F52	AS=INKEY\$
03E	0F56	IF AS=CHR\$(27) THEN GOTO 25000
03F4	0F56	
03F4	0F56	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
03F9	0F56	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
0415	0F5A	
0415	0F5A	TEXT\$="" 'CLEAR THE INPUT AREA
041E	0F5A	
041E	0F5A	GOSUB 8000 'DECODE HEADER
0423	0F5A	
0423	0F5A	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
0428	0F5A	
0428	0F5A	GOSUB 8200 'DECODE THE RESPONSE POSITIONS

```

Offset  Data    Source Line
042D 0F5A
042D 0F5A      LITHO$ = MID$(TEXT$,22,8)
043F 0F5E
043F 0F5E      GOSUB 8070 'CHECK FOR SCANNER ERRORS
0444 0F5E
0444 0F5E      GOSUB 8100 'PRINT THE DATA ON THE SCREEN
0449 0F5E
0449 0F5E      REM $INCLUDE: 'RACH930.MO1' INCLUDE THE SHORT FORM REFORMAT/EDIT MOD
0449 0F5E REM *****
0449 0F5E REM ****      AMBULATORY CARE INFORMATION SYSTEM      11 MAR 87      ****
0449 0F5E REM ****                        D R BOLLING      ****
0449 0F5E REM ****      MODULE NAME      :      RACH930.MO1      ****
0449 0F5E REM ****      SCANNER PROGRAM #      :      930-REPEAT ENCOUNTER FORM      ****
0449 0F5E REM ****
0449 0F5E REM ****      PURPOSE      :      REFORMAT/EDIT THE REPEAT      ****
0449 0F5E REM ****                        ENCOUNTER FORM.      ****
0449 0F5E REM *****
0449 0F5E REM ****      RESERVED LINE NUMBERS 100-199      ****
0449 0F5E REM *****
0449 0F5E
0449 0F5E      N.ERR = 0      'COUNTS THE NUMBER OF ERRORS
0450 0F60
0450 0F60 REM LITHO CODE DONE IN BAS PROGRAM
0450 0F60
0450 0F60 REM *** CLINIC CODE ***
0450 0F60 REM ... CLINIC CODE WHEN SINGLE BUBBLED ...
0450 0F60 100 IF MID$(TEXT$,34,2)=" " THEN 101
0468 0F60      X=VAL(MID$(TEXT$,34,2))
047E 0F62      CL1.COD$=CLIBUB$(X)
0490 0F66      GOTO 106
0494 0F66
0494 0F66 REM ... CLINIC CODE WHEN HAND CODED ...
0494 0F66 101 IF MID$(TEXT$,30,1)=" " THEN 102
04AC 0F66      IF MID$(TEXT$,31,3)=" " THEN 102
04C4 0F66      CK.X=VAL(MID$(TEXT$,30,1))
04DA 0F68      CK.COD$=MID$(TEXT$,31,3)
04EC 0F6C      CL1.COD$=CLINIC1.PFX$(CK.X) + CK.COD$
0504 0F6C      GOTO 106
0508 0F6C
0508 0F6C REM GOSUB 5500      'OUTP UCA CODE CHECK
0508 0F6C REM IF RT.5500 = 0 THEN GOTO 101
0508 0F6C 102 N.ERR = N.ERR + 1
0510 0F6C      ED.MSG$(N.ERR)="INVALID OR NO CLINIC CODE"
0524 0F6C
0524 0F6C REM *** PATIENT SSN ***
0524 0F6C 106 SSN$ = MID$(TEXT$,45,9)
0536 0F70
0536 0F70 REM *** FAMILY MEMBER PREF ***
0536 0F70 108 FMEMP$=MID$(TEXT$,54,2)
0548 0F74
0548 0F74 REM *** VISIT COUNT ***
0548 0F74 110 VCNT$="1"
0551 0F78      X$=MID$(TEXT$,56,1)
0563 0F7C      IF X$="1" THEN VCNT$="0"

```

Offset	Data	Source Line
057A	0F7C	IF X\$>"1" THEN VCNTS=X\$
0591	0F7C	
0591	0F7C	REM *** PROVIDER TIME ***
0591	0F7C	112 X=VAL(MID\$(TEXT\$,57,2))
05A7	0F7C	PPROV.TIMS=PROV.TIMES(X)
05B9	0F80	
05B9	0F80	REM *** REFERRAL CODE ***
05B9	0F80	CK.X=VAL(MID\$(TEXT\$,59,1))
05CF	0F80	CK.CODS=MID\$(TEXT\$,60,3)
05E1	0F80	REM IF CK.X=0 AND CK.CODS=" " THEN GOTO 114 'MAY NOT BE CODED
05E1	0F80	REM GOSUB 5600 'VALIDATE INP UCA CODE
05E1	0F80	REM IF RT.5600 = 0 THEN GOTO 113
05E1	0F80	REM N.ERR = N.ERR + 1
05E1	0F80	REM ED.MSG\$(N.ERR)="#11-12 INVALID REFERRAL CODE"
05E1	0F80	
05E1	0F80	113 REF.PFX\$=REFER.PFX\$(CK.X)
05F3	0F84	REF.CODS=CK.CODS
05FC	0F88	
05FC	0F88	REM *** PROCEDURE CODE ***
05FC	0F88	114 PROC\$=MID\$(TEXT\$,63,5)
060E	0F8C	
060E	0F8C	REM *** DIAGNOSIS ***
060E	0F8C	
060E	0F8C	116 X=VAL(MID\$(TEXT\$,68,1))
0624	0F8C	IF X=0 THEN PRIMDX\$=""
0638	0F90	IF X=1 THEN PRIMDX\$="V"
064C	0F90	IF X=2 THEN PRIMDX\$="S"
0660	0F90	DXTMP\$=MID\$(TEXT\$,69,5)
0672	0F94	IF DXTMP\$=" " THEN GOTO 120
0684	0F94	
0684	0F94	REM REMOVE LEADING BLANKS
0684	0F94	118 IF LEFT\$(DXTMP\$,1)=" " THEN DXTMP\$=RIGHT\$(DXTMP\$,4)+" ":GOTO 118
06B3	0F94	
06B3	0F94	120 LASTCS=RIGHT\$(DXTMP\$,1) 'GET LAST CHAR
06C2	0F98	IF LASTCS<>" " AND LASTCS<>"0" THEN PRIMDX\$="" 'REMOVE V OR S
06EE	0F98	PRIMDX\$=LEFT\$(PRIMDX\$+DXTMP\$,5) 'GET 5 CHAR ONLY
0700	0F98	
0700	0F98	IF PRIMDX\$<>" " THEN 121
070E	0F98	N.ERR=N.ERR+1
0716	0F98	ED.MSG\$(N.ERR)="NO DIAGNOSIS CODED"
072A	0F98	
072A	0F98	121 REM
072B	0F98	VDAT.OFFSET = 74
072C	0F9A	PPFX.OFFSET = 78
0739	0F9C	PNUM.OFFSET = 79
0740	0F9E	TOT.OFFSET = 9
0747	0FA0	POINTER = 0
074E	0FA2	
074E	0FA2	CM\$=MID\$(DATE\$,1,2) 'CURRENT MONTH
0760	0FA6	
0760	0FA6	REM LOOP THROUGH THE FOLLOWING CODE 5 TIMES
0760	0FA6	FOR I421 = 1 TO 5
0767	0FA6	REM *** VISIT DATE ***
0767	0FA6	YR\$=MID\$(DATE\$,9,2) 'CURRENT YEAR

Offset Data Source Line

```

0779 OFAA      XS=MID$(TEXT$,VDAT.OFFSET+POINTER,4)  'MONTH AND DAY FROM FORM
0790 OFAA      IF XS=STRING$(4," ") THEN GOTO 128
07A9 OFAA      IF LEFT$(XS,2)<=CM$ THEN 122      'OK, USE THIS YEAR
07BE OFAA      YRS=RIGHT$(STR$(VAL(YRS)-1),2)  'USE LAST YEAR
07DE OFAA      122 VIDATES=YRS + XS
07EC OFAE      D1$=MID$(XS,3,1)
07FE OFB2      D2$=MID$(XS,4,1)
0810 OFB6      IF D1$=" " AND D2$<>" " THEN 123
0833 OFB6      IF D1$<>" " AND D2$=" " THEN 123
0856 OFB6      REM ... EDIT VISIT DATE ...
0856 OFB6      CK.5000$=VIDATES
085F OFBA      GOSUB 5000      'DATE CHECK
0864 OFBA      IF RT.5000=0 THEN GOTO 124
0873 OFBC      123 N.ERR=N.ERR+1
087B OFBC      ED.MSG$(N.ERR)="VISIT DATE ERROR IN ITEM "+STR$(I421)
0898 OFBE
0898 OFBE      REM CARE PROVIDER
0898 OFBE      124 CPROV.PFX$=MID$(TEXT$,PPFX.OFFSET+POINTER,1)
08AF OFC2      CPROV.NUM$=MID$(TEXT$,PNUM.OFFSET+POINTER,4)
08C6 OFC6
08C6 OFC6      126 PVDC$(I421)=VIDATES+CPROV.PFX$+CPROV.NUM$
08E5 OFC6      GOTO 130
08E9 OFC6
08E9 OFC6      128 PVDC$(I421)=STRING$(11," ")
0905 OFC6
0905 OFC6      130 REM
0906 OFC6      POINTER = POINTER + TOT.OFFSET
0911 OFC6      NEXT I421
0923 OFC6      199 REM
0924 OFC6
0924 OFC6      REM -----END OF MODULE RACH930.M01-----
0924 OFC6
0924 OFC6      IF N.ERR = 0 THEN GOTO 997
0933 OFC6      LPRINT "LITHO # ";LITHO$;" ... ERRORS"
0945 OFC6      FOR I997 = 1 TO N.ERR
0952 OFC8          LPRINT USING "### ";I997;
095E OFCA          LPRINT "==> ";ED.MSG$(I997)
0974 OFCA      NEXT I997
0985 OFCA      LN.COUNT = LN.COUNT + N.ERR + 1
0991 OFCC      CNTRLOPT = 6
0998 OFCC      GOSUB 9010      'REJECT THE FORM
099D OFCC      GOTO 998      'BYPASS THE DISK WRITER....
09A1 OFCC
09A1 OFCC      997 REM $INCLUDE: 'RACH930.M02' REM INCLUDE THE BASE ENCOUNTER FORM DISK WRITER
09A2 OFCC      REM *****
09A2 OFCC      REM ***** AMBULATORY CARE INFORMATION SYSTEM 11 MAR 87 *****
09A2 OFCC      REM ***** D R BOLLING *****
09A2 OFCC      REM ***** MODULE NAME : RACH930.M02 *****
09A2 OFCC      REM ***** SCANNER PROGRAM # : 930-REPEAT ENCOUNTER FORM *****
09A2 OFCC      REM ***** PURPOSE : CREATE AND WRITE THE DISK *****
09A2 OFCC      REM ***** RECORD FOR INPUT TO FOCUS *****
09A2 OFCC      REM *****

```

```

Offset  Data   Source Line
09A2 0FCC REM *****
09A2 0FCC REM ****  RESERVED LINE NUMBERS 200-299  ****
09A2 0FCC REM *****
09A2 0FCC
09A2 0FCC REM BUILD THE OUTPUT RECORD
09A2 0FCC
09A2 0FCC GOSUB 270 'BUILD FIRST PART OF THE RECORD KEY (RECKEY$)
09A7 0FCC
09A7 0FCC REM *****
09A7 0FCC REM ****  RECORD TYPE "1" - MAIN TRANSACTION  ****
09A7 0FCC REM *****
09A7 0FCC
09A7 0FCC FOR I412 = 1 TO 5
09AE 0FCC IF PVDC$(I412)=STRING$(11," ") THEN GOTO 250
09D4 0FCE RECOUT$ ="9301"+RECKEY$+PVDC$(I412) 'BEGIN KEY WITH VISIT DATE
09F5 0FD6 ISTEP = I412
09FC 0FD8 GOSUB 272 'DO SECOND PART OF KEY
0A01 0FD8 GOSUB 274 'DO TYPE 1 RECORD
0A06 0FD8 RECOUT$ =RECOUT$+RECKEY2$+RECTYP1$ 'ADD TYPE 1 RECORD
0A19 0FE0 GOSUB 280 'PAD
0A1E 0FE0 PRINT #1,RECOUT$ 'WRITE IT
0A29 0FE0 GOSUB 276 'DO TYPE 2 RECORD
0A2E 0FE0 GOSUB 280 'PAD
0A33 0FE0 PRINT #1,RECOUT$ 'WRITE IT
0A3E 0FE0 250 NEXT I412
0A50 0FE0
0A50 0FE0 GOTO 299
0A54 0FE0
0A54 0FE0 REM *****
0A54 0FE0 REM ****  SUBROUTINE 270 - BUILD THE RECORD KEY PART 1  ****
0A54 0FE0 REM *****
0A54 0FE0 270 RECKEY$=""
0A5D 0FE0
0A5D 0FE0 REM *** CLINIC ID (PREFIX + COD) ***
0A5D 0FE0 RECKEY$= CL1.PFX$+CL1.COD$
0A6B 0FE4
0A6B 0FE4 RETURN
0A6E 0FE4
0A6E 0FE4 REM *** RECORD KEY PART 2 - CALLED WITHIN LOOP TO ALTER LITHO- ***
0A6E 0FE4
0A6E 0FE4 272 RECKEY2$=""
0A77 0FE4
0A77 0FE4 REM *** PATIENT SSN ***
0A77 0FE4 RECKEY2$ = SSN$
0A80 0FE4
0A80 0FE4 REM *** FAMILY MEMBER PREF ***
0A80 0FE4 RECKEY2$ = RECKEY2$ + FMEMP$
0A8C 0FE4
0A8C 0FE4 REM *** LITHO CODE ***
0A8C 0FE4 RECKEY2$ = RECKEY2$ + RIGHT$(STR$(ISTEP),1) + RIGHT$(LITHO$,7)
0A81 0FE4
0A81 0FE4 REM *** FORM NUMBER ***
0A81 0FE4 RECKEY2$ = RECKEY2$ + "93"
0A8D 0FE4

```

Offset	Data	Source Line
0A80	0FE4	RETURN
0AC0	0FE4	
0AC0	0FE4	REM *****
0AC0	0FE4	REM **** SUBROUTINE 275 - BUILD RECORD TYPE 1 ****
0AC0	0FE4	REM *****
0AC0	0FE4	274 RECTYP1\$=""
0AC9	0FE4	
0AC9	0FE4	REM *** VISIT COUNT ***
0AC9	0FE4	RECTYP1\$ = RECTYP1\$ + VCNT\$
0AD5	0FE4	
0AD5	0FE4	REM *** PROV TIME ***
0AD5	0FE4	RECTYP1\$ = RECTYP1\$ + PPROV.TIMS
0AE1	0FE4	
0AE1	0FE4	REM *** SPACES ***
0AE1	0FE4	RECTYP1\$ = RECTYP1\$ + STRING\$(10," ")
0AF3	0FE4	
0AF3	0FE4	REM *** REFERRAL CODE (PREFIX AND COD) ***
0AF3	0FE4	RECTYP1\$ = RECTYP1\$ + REF.PFX\$ + REF.COD\$
0B06	0FE4	
0B06	0FE4	REM *** SPACES ***
0B06	0FE4	RECTYP1\$ = RECTYP1\$ + STRING\$(11," ")
0B18	0FE4	
0B18	0FE4	REM *** PATIENT DIAGNOSIS (PREFIX AND NUM) ***
0B18	0FE4	RECTYP1\$ = RECTYP1\$ + PRIMDX\$
0B24	0FE4	
0B24	0FE4	RETURN
0B27	0FE4	
0B27	0FE4	REM *****
0B27	0FE4	REM *** RECORD TYPE "2" - RECKEY + PROCEDURE CODE ***
0B27	0FE4	REM *****
0B27	0FE4	276 REM
0B28	0FE4	
0B28	0FE4	RECOU\$ ="9302"+RECKEY\$+PVDC\$(I412)+RECKEY2\$ 'BUILD KEY
0B4F	0FE4	RECOU\$ = RECOU\$ + "1" + PROC\$ 'ADD PROCEDURE
0B62	0FE4	
0B62	0FE4	RETURN
0B65	0FE4	
0B65	0FE4	REM *****
0B65	0FE4	REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
0B65	0FE4	REM *****
0B65	0FE4	280 PAD=MAXLENGTH - LEN(RECOU\$) 'FIND OUT HOW SHORT THE RECORD IS
0B73	0FE6	RECOU\$ = RECOU\$ + STRING\$(PAD,PAD\$) 'PAD THE RECORD WITH FILL CHAR
0B86	0FE6	RETURN
0B89	0FE6	
0B89	0FE6	299 REM
0B8A	0FE6	
0B8A	0FE6	REM -----END OF MODULE RXXM930.MO2-----
0B8A	0FE6	
0B8A	0FE6	998 REM CONTINUE
0B8B	0FE6	
0B8B	0FE6	999 READTYPE = 2
0B92	0FE6	IF LN.COUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
0BA2	0FE6	GOTO 10
0BA6	0FE6	

```

Offset  Data   Source Line
-----  -
0BA6  OFE6  REM  END OF SCAN/DECODE/WRITE LOOP =====
0BA6  OFE6
0BA6  OFE6  1000  REM $INCLUDE: 'RACS1000.SUB' Include the VERIFY LOGON SUB
0BA7  OFE6  REM *****
0BA7  OFE6  REM * NAME: RACS1000          LOGON VERIFICATION SUBROUTINE      *
0BA7  OFE6  REM * Date: 28 Feb 84        PATIENT REGISTRATION PROGRAM      *
0BA7  OFE6  REM *****
0BA7  OFE6  REM          PATIENT OMR INPUT PROGRAM                          *
0BA7  OFE6  REM                                                                 *
0BA7  OFE6  REM This program verifies user is logged on properly. If there is no *
0BA7  OFE6  REM valid user logged on at the time of execution, this subroutine will*
0BA7  OFE6  REM chain to the logon program RACP05, otherwise a return is issued.  *
0BA7  OFE6  REM *****
0BA7  OFE6  REM  RESERVED LINE NUMBERS ARE 1001 THRU 1010
0BA7  OFE6  REM *****
0BA7  OFE6  1001  OPEN "I",1,"RACLOG.DAT"
0BB9  OFE6          IF EOF(1) THEN 1002          'MAKE THEM LOG ON FIRST
0BC7  OFE6          INPUT #1,USERS(1),DTS,TMS,PIDS
0BE8  OFF2          IF USERS(1) = "" THEN 1002          'MAKE THEM LOG ON FIRST
0BF6  OFF2          IF USERS(1) = "*****" THEN 1002      'MAKE THEM LOG ON FIRST
0C04  OFF2          CLOSE 1
0C08  OFF2          SCREEN 0,1,0,0
0C21  OFF2          COLOR FORE,BACK,BORD
0C37  OFF2          CLS
0C38  OFF2          RETURN
0C3E  OFF2
0C3E  OFF2  1002  CLOSE
0C42  OFF2          CHAIN "RACP05"
0C49  OFF2  '=====END OF LOGON VERIFY SUBROUTINE 1000=====
0C49  OFF2
0C49  OFF2  2000  REM $INCLUDE: 'RACS2000.SUB' Include the REPLY/DELAY SUB
0C4A  OFF2  REM *****
0C4A  OFF2  REM *****  AMBULATORY CARE DATA BASE          13 APR 85      ****
0C4A  OFF2  REM *****                                SKIP COLE          ****
0C4A  OFF2  REM *****  SUBROUTINE NAME      :  RACS2000.SUB          ****
0C4A  OFF2  REM *****  SCANNER PROGRAM #    :  ALL                  ****
0C4A  OFF2  REM *****  FUNCTION              :  THIS SUBROUTINE MODULE ****
0C4A  OFF2  REM *****                                SERVERS AS A WAIT AND REPLY ****
0C4A  OFF2  REM *****                                ENTRY MODULE          ****
0C4A  OFF2  REM *****  INPUT                  :  SINGLE KEYBOARD ENTRY ****
0C4A  OFF2  REM *****                                ****
0C4A  OFF2  REM *****  OUTPUT                  :  KEYBOARD ENTRY - UPPER CASE ****
0C4A  OFF2  REM *****                                ****
0C4A  OFF2  REM *****  RESERVED LINE          ****
0C4A  OFF2  REM *****  NUMBERS              :  2001-2010          ****
0C4A  OFF2  REM *****
0C4A  OFF2  2001  REM REPLY FUNCTION
0C4B  OFF2  2002  REPLY$=INKEY$ : IF REPLY$="" THEN 2002
0C5F  OFF6          REPLY=ASC(REPLY$)
0C61  OFF8          IF REPLY > 90 THEN REPLY$=CHR$(REPLY XOR 32) 'CONVERT TO CAPS
0CB4  OFF8          IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$="?"
0CB0  OFF8          RETURN
0CB3  OFF8
0CB3  OFF8  5000  REM $INCLUDE: 'RACS5000.SUB' Include the DATE EDITOR SUB

```

Offset	Data	Source	Line	
0CB4	OFFB	REM	*****	
0CB4	OFFB	REM	**** AMBULATORY CARE DATA BASE	13 APR 85 ****
0CB4	OFFB	REM	****	SKIP COLE ****
0CB4	OFFB	REM	**** SUBROUTINE NAME :	RXXS5000.SUB ****
0CB4	OFFB	REM	**** SCANNER PROGRAM # :	ALL ****
0CB4	OFFB	REM	**** FUNCTION :	THIS SUBROUTINE MODULE ****
0CB4	OFFB	REM	****	PERFORMS A DATE EDIT ****
0CB4	OFFB	REM	****	****
0CB4	OFFB	REM	**** INPUT :	DATE TO BE CHECKED MUST BE ****
0CB4	OFFB	REM	****	IN THE VARIABLE NAMED ****
0CB4	OFFB	REM	****	'CK.5000\$' ****
0CB4	OFFB	REM	****	IN THE FORMAT "YYMMDD" ****
0CB4	OFFB	REM	****	****
0CB4	OFFB	REM	**** OUTPUT :	'RT.5000' IS THE RETURN CODE ****
0CB4	OFFB	REM	****	VARIABLE. IF THIS VARIABLE ****
0CB4	OFFB	REM	****	CONTAINS ANY NUMBER OTHER ****
0CB4	OFFB	REM	****	THAN 0, AN ERROR WAS FOUND ****
0CB4	OFFB	REM	****	IN THE DATE. ****
0CB4	OFFB	REM	****	****
0CB4	OFFB	REM	**** RESERVED LINE	****
0CB4	OFFB	REM	**** NUMBERS :	5001-5009 ****
0CB4	OFFB	REM	*****	
0CB4	OFFB		RT.5000 = 0	
0CB8	OFFB		CKYEAR = VAL(LEFT\$(CK.5000\$,2))	'YEAR NUMERIC VALUE
0CCE	OFFA		CKMONTH = VAL(MID\$(CK.5000\$,3,2))	'MONTH NUMERIC VALUE
0CE4	OFFC		CKDAY = VAL(RIGHT\$(CK.5000\$,2))	'DAY NUMERIC VALUE
0CF7	OFFE			
0CF7	OFFE		IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009	
0D0D	OFFE		IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009	
0D23	OFFE		IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009	
0D39	OFFE		IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009	
0D4F	OFFE			
0D4F	OFFE	REM	LEAP YEAR CHECK	
0D4F	OFFE		MOLENGTH(2) = 28	
0D56	OFFE		IF CKMONTH<> 2 THEN GOTO 5005	'MUST BE FEBRUARY
0D65	OFFE		IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005	'MUST BE A LEAP YEAR
0D7A	OFFE		MOLENGTH(2) = 29	
0D81	OFFE			
0D81	OFFE	5005	IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009	
0DA0	OFFE			
0DA0	OFFE	5009	RETURN	
0DA3	OFFE			
0DA3	OFFE	REM	-----END OF SUBROUTINE 5000 -----	
0DA3	OFFE			
0DA3	OFFE	5010	REM \$INCLUDE: 'RACS5010.SUB' Include the NUMERIC STRING EDITOR	
0DA4	OFFE	REM	*****	
0DA4	OFFE	REM	**** AMBULATORY CARE DATA BASE	1 MAY 85 ****
0DA4	OFFE	REM	****	SKIP COLE ****
0DA4	OFFE	REM	**** SUBROUTINE NAME :	RXXS5010.SUB ****
0DA4	OFFE	REM	**** SCANNER PROGRAM # :	ALL ****
0DA4	OFFE	REM	**** FUNCTION :	THIS SUBROUTINE MODULE ****
0DA4	OFFE	REM	****	PERFORMS A NUMERIC STRING ****
0DA4	OFFE	REM	****	EDIT. ****
0DA4	OFFE	REM	****	****

```

Offset  Data  Source Line
-----
00A4  OFFE  REM  ****  INPUT          :  STRING TO BE EDITED IS IN  ****
00A4  OFFE  REM  ****  THE VARIABLE NAMED  ****
00A4  OFFE  REM  ****  'CK.5010$'  ****
00A4  OFFE  REM  ****  ****
00A4  OFFE  REM  ****  OUTPUT          :  'RT.5010' IS THE RETURN CODE  ****
00A4  OFFE  REM  ****  VARIABLE. IF THIS VARIABLE  ****
00A4  OFFE  REM  ****  CONTAINS ANY NUMBER OTHER  ****
00A4  OFFE  REM  ****  THAN 0, AN ERROR WAS FOUND  ****
00A4  OFFE  REM  ****  IN THE STRING.  ****
00A4  OFFE  REM  ****  ****
00A4  OFFE  REM  ****  RESERVED LINE  ****
00A4  OFFE  REM  ****  NUMBERS          :  5011-5019  ****
00A4  OFFE  REM  ****  ****
00A4  OFFE  REM  ****  RT.5010 = 0  ****
00AB  1000
00AB  1000  FOR I5010 = 1 TO LEN(CK.5010$)
00BB  1006  J5010= ASC(MID$(CK.5010$,I5010,1))
00CF  100A  IF J5010 < 48 OR J5010 > 57 THEN RT.5010 = RT.5010 + 1
00F7  100A  NEXT I5010
0E08  100A
0E08  100A  RETURN
0E08  100A  REM  ----- END OF SUBROUTINE 5010 -----
0E08  100A  5500  REM  INCLUDE: 'RACS5500.SUB' Include the OUTPATIENT UCA CHECK SUB
0E0C  100A  5600  REM  INCLUDE: 'RACS5600.SUB' Include the INPATIENT UCA CHECK SUB
0E0D  100A
0E0D  100A  7000  REM $INCLUDE: 'RACS7000.SUB' Include the SCREEN HEADER SUB
0E0E  100A  REM  ****
0E0E  100A  REM  ****  AMBULATORY CARE DATA BASE 13 APR 85 ****
0E0E  100A  REM  ****  SKIP COLE ****
0E0E  100A  REM  ****  SUBROUTINE NAME      :  RACS7000.SUB ****
0E0E  100A  REM  ****  SCANNER PROGRAM #    :  ALL ****
0E0E  100A  REM  ****  FUNCTION          :  THIS SUBROUTINE MODULE ****
0E0E  100A  REM  ****  PRINTS THE STANDARD SCREEN ****
0E0E  100A  REM  ****  HEADING. ****
0E0E  100A  REM  ****  INPUT          :  COMMON VARIABLE USER$(2) ****
0E0E  100A  REM  ****  SYSTEM DATE ****
0E0E  100A  REM  ****  ****
0E0E  100A  REM  ****  OUTPUT          :  SCREEN HEADING ****
0E0E  100A  REM  ****  ****
0E0E  100A  REM  ****  RESERVED LINE ****
0E0E  100A  REM  ****  NUMBERS          :  7001-7010 ****
0E0E  100A  REM  ****  ****
0E0E  100A  REM  ****
0E0E  100A  7001  LOCATE 1,1
0E18  100A  PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
0E20  100A  LOCATE 1,65
0E2D  100A  PRINT DATES;
0E35  100A  LOCATE 2,1
0E42  100A  PRINT "USER : ";USER$(1)
0E4F  100A  RETURN
0E52  100A  7100  REM $INCLUDE: 'RACS7100.SUB' Include the PRINTER HEADER SUB
0E53  100A  REM  ****
0E53  100A  REM  ****  AMBULATORY CARE DATA BASE 13 APR 85 ****
0E53  100A  REM  ****  SKIP COLE ****

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

0E53 100A REM **** SUBROUTINE NAME      :  RXXS7100.SUB      ****
0E53 100A REM **** SCANNER PROGRAM #    :  ALL              ****
0E53 100A REM **** FUNCTION              :  THIS SUBROUTINE MODULE ****
0E53 100A REM ****                      :  PRINTS THE STANDARD HEADING ****
0E53 100A REM ****                      :  ON THE PRINTER.      ****
0E53 100A REM **** INPUT                  :  DATE,PAGE,PGMIDS,PGMTITLS ****
0E53 100A REM ****                      :                      ****
0E53 100A REM **** OUTPUT                  :  PRINTER HEADING, LN.COUNT ****
0E53 100A REM ****                      :                      ****
0E53 100A REM **** RESERVED LINE          ****
0E53 100A REM **** NUMBERS                :  7101-7110        ****
0E53 100A REM ****                      ****
0E53 100A
0E53 100A 7101 IF PAGE > 0 THEN LPRINT CHR$(12);
0E69 100A      LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITLS;
0E76 100A      LPRINT TAB(70);DATES
0E89 100A      PAGE=PAGE+1
0E91 100A      LPRINT "PROGRAM ";PGMIDS;TAB(70);"PAGE";
0EAE 100A      LPRINT USING "####";PAGE
0EBA 100A      LPRINT
0EC2 100A      LN.COUNT=3
0EC9 100A      RETURN
0ECC 100A
0ECC 100A 8000 REM $INCLUDE: 'RACS8000.SUB' Include the DECODE SUB GROUP
0ECD 100A REM ****
0ECD 100A REM **** AMBULATORY CARE DATA BASE      13 APR 85    ***
0ECD 100A REM ****                      SKIP COLE          ***
0ECD 100A REM **** SUBROUTINE NAME      :  RXXS8000.SUB      ***
0ECD 100A REM **** SCANNER PROGRAM #    :  ALL              ***
0ECD 100A REM **** FUNCTION              :  THIS SUBROUTINE MODULE ****
0ECD 100A REM ****                      :  IS A GROUPING THAT PERFORMS ****
0ECD 100A REM ****                      :  VARIOUS DECODING FUNCTIONS ****
0ECD 100A REM ****                      :  ON THE SCANNER DATA ****
0ECD 100A REM ****                      ****
0ECD 100A REM **** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ****
0ECD 100A REM **** 8050 - CHECK FOR END OF JOB ****
0ECD 100A REM **** 8100 - PRINT THE HEADER DATA ON THE SCREEN ****
0ECD 100A REM **** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-..) ****
0ECD 100A REM ****                      (RETURNED IN TEXT$ STRING VARIABLE) ****
0ECD 100A REM ****                      ****
0ECD 100A REM **** INPUT                  :  SHEET RECORD, RECORD LENGTH ****
0ECD 100A REM ****                      ****
0ECD 100A REM **** OUTPUT                  :  'TEXT$' TRING VARIABLE ****
0ECD 100A REM ****                      ****
0ECD 100A REM **** RESERVED LINE          ****
0ECD 100A REM **** NUMBERS                :  8001-8500        ****
0ECD 100A REM ****                      ****
0ECD 100A
0ECD 100A 'DECODE THE HEADER ONLY
0ECD 100A 8001  POINTER = 0
0ED4 100A      RECORDPTR = VARPTR(SHEETREC(0))
0ED8 100C      FOR J8000 = 1 TO 21
0EE2 100C 8002  TEXT$= TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
0F00 100C      POINTER=POINTER+1

```

Offset	Data	Source Line
0F08	100C	NEXT J8000
0F17	100E	PROGRAM\$= LEFT\$(TEXT\$,3)
0F26	1012	BATCH\$= MID\$(TEXT\$,4,3)
0F38	1016	SERIAL\$= MID\$(TEXT\$,7,4)
0F4A	101A	RUNID\$= MID\$(TEXT\$,11,1)
0F5C	101E	FORM\$= MID\$(TEXT\$,12,2)
0F6E	1022	POCKET\$= MID\$(TEXT\$,14,1)
0F80	1026	SCANERR1\$=MID\$(TEXT\$,16,2)
0F92	102A	SCANERR2\$=MID\$(TEXT\$,18,2)
0FA4	102E	SCANERR3\$=MID\$(TEXT\$,20,2)
0FB6	1032	GOTO 8500
0FBA	1032	
0FBA	1032	8050 REM CHECK FOR END OF JOB/END OF BATCH
0FBB	1032	IF PROGRAM\$ = PGMID\$ THEN GOTO 8500
0FCD	1032	LPRINT STRING\$(80,"")
0FDB	1032	LPRINT
0FE3	1032	LPRINT "RECORDS PROCESSED ... ";SERIAL\$
0FF0	1032	LPRINT "STARTED AT ";BTIM\$
0FFD	1032	LPRINT "ENDED AT ";TIM\$
100A	1032	LPRINT CHR\$(12)
1015	1032	GOTO 30000
1019	1032	
1019	1032	8070 REM CHECK FOR SCANNER ERRORS
101A	1032	IF POCKET\$ = " " GOTO 8500
102C	1032	LPRINT LITHOS;
1034	1032	LPRINT " ... SCANNER ERRORS : ";
103C	1032	LPRINT SCANERR1\$;" / ";
1049	1032	LPRINT SCANERR2\$;" / ";
1056	1032	LPRINT SCANERR3\$
105E	1032	LN=LN+1
1066	1034	GOTO 999
106A	1034	
106A	1034	8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
106B	1034	LOCATE 5,1:PRINT "PROGRAM ";PROGRAM\$;
1085	1034	
1085	1034	PRINT " BATCH ";BATCH\$;
1092	1034	PRINT " RUN ";RUNID\$;
109F	1034	PRINT " FORM ";FORM\$;
10AC	1034	PRINT " POCKET ";POCKET\$
10B9	1034	GOTO 8500
10BD	1034	
10D	1034	8200 REM DECODE THE RESPONSE POSITIONS
10BE	1034	POINTER = 21
10C5	1034	RECORDPTR = VARPTR(SHEETREC(0))
10CC	1034	FOR J8000 = 22 TO RECORDLENGTH
10D9	1036	8202 TEXT\$ = TEXT\$+CHR\$(PEEK(RECORDPTR + POINTER))
10F7	1036	POINTER=POINTER+1
10FF	1036	NEXT J8000
1110	1036	
1110	1036	8500 RETURN
1113	1036	
1113	1036	REM ----- END OF RXS8000.SUB -----
1113	1036	
1113	1036	9000 REM \$INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

1114 1036 REM *****
1114 1036 REM ****    AMBULATORY CARE DATA BASE          13 APR 85    ****
1114 1036 REM ****                                SKIP COLE      ****
1114 1036 REM ****    PROGRAM NAME      :    RACS9000.SUB          ****
1114 1036 REM ****    SCANNER PROGRAM #  :    ALL                  ****
1114 1036 REM ****    FUNCTION            :    THIS SUBROUTINE MODULE ****
1114 1036 REM ****                                CONTROLS THE SCANNER I/O ****
1114 1036 REM ****                                ****
1114 1036 REM ****    INPUT/OUTPUT          :    REFER TO THE ASYNCHRONOUS ****
1114 1036 REM ****                                COMMUNICATIONS MANUAL AND THE ****
1114 1036 REM ****                                PRE-RELEASED SOFTWARE GUIDE ****
1114 1036 REM ****                                ****
1114 1036 REM *****
1114 1036 REM ****    RESERVED LINE                                ****
1114 1036 REM ****                                NUMBERS      :    9001-9100 ****
1114 1036 REM *****
1114 1036
1114 1036
1114 1036 REM *****
1114 1036 REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
1114 1036 REM **** ARGUMENTS: PRESET ... SEE BELOW ****
1114 1036 REM *****
1114 1036 9001 REM
1115 1036          PROTOCOL(0) = 9600          'BAUD RATE
1116 1036          PROTOCOL(1) = 78            'PARITY (SEE PAGE 4-8 OF MANUAL)
1123 1036          PROTOCOL(2) = 8            'DATA BITS
112A 1036          PROTOCOL(3) = 1            'STOP BITS
1131 1036          PROTOCOL(4) = 2            'RS-232 PORT
1138 1036          PROTOCOL(5) = 0            'WRITE TIME-OUT
113F 1036          PROTOCOL(6) = 0            'READ TIME-OUT
1146 1036
1146 1036          ERRSTAT$ = SPACE$(60)
1152 1036          ARGPTR = VARPTR(PROTOCOL(0))
1159 1038          CALL SETUP (ARGPTR,ERRSTAT$)
116A 1038          ERRMSG$=""
1173 1038          IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="SETUP ERROR "+ERRSTAT$
118F 1038          GOTO 9100
1193 1038
1193 1038 REM *****
1193 1038 REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
1193 1038 REM **** ARGUMENTS: CNTRLOPT ****
1193 1038 REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
1193 1038 REM **** CNTRLOPT = 2 = STOP SCANNER (S0) ****
1193 1038 REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
1193 1038 REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
1193 1038 REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
1193 1038 REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
1193 1038 REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
1193 1038 REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
1193 1038 REM *****
1193 1038 9010 REM
1194 1038          ERRSTAT$ = SPACE$(60)
11A0 1038          CALL CNTRL (CNTRLOPT,ERRSTAT$)
11B1 1038          ERRMSG$=""

```

```

Offset  Data    Source Line
118A 1038      IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="CONTROL ERROR "+ERRSTAT$
11D6 1038      GOTO 9100
11DA 1038
11DA 1038 REM *****
11DA 1038 REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
11DA 1038 REM ****
11DA 1038 REM **** ARGUMENTS: READTYPE ****
11DA 1038 REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
11DA 1038 REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
11DA 1038 REM ****
11DA 1038 REM **** ARGUMENTS: RECORDLENGTH ****
11DA 1038 REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
11DA 1038 REM **** TRANSMITTED ****
11DA 1038 REM *****
11DA 1038 9020 REM
11D8 1038      ERRSTAT$ = SPACE$(60)
11E7 1038      RECORDPTR = VARPTR(SHEETREC(0))
11EE 1038      CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT$)
1207 1038      ERRMSG$=""
1210 1038      IF MID$(ERRSTAT$,14,3) = "415" THEN ERRMSG$="ESC"
1231 1038      GOTO 9100
1235 1038
1235 1038 REM *****
1235 1038 REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
1235 1038 REM ****
1235 1038 REM **** ARGUMENTS: PRINTPOS ****
1235 1038 REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
1235 1038 REM **** VALUES = 0 THRU 90 ****
1235 1038 REM ****
1235 1038 REM **** ARGUMENTS: PSTRINGS ****
1235 1038 REM **** TEXT TO BE PRINTED ON THE FORM ****
1235 1038 REM ****
1235 1038 REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
1235 1038 REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
1235 1038 REM *****
1235 1038 9030 REM
1236 1038      ERRSTAT$ = SPACE$(60)
1242 1038      RECORDPTR = VARPTR(SHEETREC(0))
1249 1038      CALL TPRINT(PRINTPOS,PSTRINGS$,ERRSTAT$)
125E 103E      IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="PRINT ERROR "+ERRSTAT$
127A 103E      GOTO 9100
127E 103E
127E 103E 9100 RETURN
1281 103E REM -----END OF SUBROUTINE RACS9000.SUB -----
1281 103E
1281 103E REM END OF SUBROUTINES =====
1281 103E
1281 103E 25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
1282 103E      LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
128A 103E      LPRINT "ERASING FILE ";DATFIL$
1297 103E      BEEP
1298 103E      CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
12A7 103E      CLOSE
12AB 103E      OPEN DATFIL$ FOR OUTPUT AS #1

```

RACP930

REPEAT ENCOUNTER DESTING/DECODE PROGRAM

PAGE 19

07-06-87

09:25:22

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

128D	103E	PRINT #1,STRINGS(RECORDLENGTH,"X")	'VOID THE FIRST RECORD
12CF	103E	CLOSE	
12D3	103E		
12D3	103E	30000 REM	
12D4	103E	CLOSE	
12D8	103E	CHAIN "RACP10"	
12DF	103E		
12E2	103E		

22151 Bytes Available

17154 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

PAGE 1

07-06-87

15:07:19

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

001A 0002 REM \$LINESIZE: 132
001A 0002 REM \$PAGESIZE: 66
001A 0002 REM \$TITLE: 'RACP940 '
001A 0002 REM \$SUBTITLE: 'SHORT FORM'
001A 0002 REM \$PAGE

RACP940
SHORT FORM

PAGE 2
07-06-87
15:07:19

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```
001A 0002 REM +-----+
001A 0002 REM | NAME: RACP940          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | DATE: 19 MAR 87                                |
001A 0002 REM | D R BOLLING          SHORT FORM                    |
001A 0002 REM |
001A 0002 REM | INCLUDES PREFIX TO LITHO FOR EACH PATIENT ON FORM |
001A 0002 REM +-----+
001A 0002 REM          SHORT FORM INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the SHORT form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RXXP05, otherwise,
001A 0002 REM the program chains to program RXXP10 on exit.
001A 0002
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD          DIMENSION DEFINITIONS      *
001A 0002 '* Date: 28 Feb 84          Written by: Floyd Cole      *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002
001A 0002 DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 DIM ED.MSG$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002 DIM CLINIC1.PFX$(5) '(PREFIX -B D F G S- FOR CLINIC #1)
001A 0002 DIM PROVIDER.TIMES(9) '(TIME TABLE FOR PROVIDERS)
001A 0002 DIM PROCED$(16) '(PROCEDURE TABLE FOR SHORT FORM)
```

Offset	Data	Source Line
001A	0002	DIM PTID\$(5) '(PATIENT SSN+FMP)
001A	0002	DIM PTIM\$(5) '(PATIENT TIME)
001A	0002	DIM TOT.PROC(5) '(PROCEDURE TOTAL PER PATIENT)
001A	0002	DIM GROUP\$(5,16) '(PATIENT/PROCEDURE GROUP)
001A	0002	DIM SPE.COD\$(5) '(SPECIFIC PRE CODE GROUP)
001A	0002	DIM HOLD\$(16) '(HOLD AREA FOR SUBROUTINE 6000)
001A	0002	
001A	0002	REM \$INCLUDE: 'RACCMN.MOD' REM INCLUDE THE COMMON AREA DEFINITION
001A	0002	*****
001A	0002	* NAME: RACCMN.MOD COMMON AREA DEFINITION *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A	0002	' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	' *****START OF COMMON DEFINITIONS*****
001A	0002	
001A	0002	COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL\$ 'BASIC SCREEN COLORS
001A	0002	COMMON HEADERS\$ '21 CHARACTER SCANNER HEADER INFO
001A	0002	COMMON TEXT\$ '11 AINING CHARACTERS FROM SCANNER
001A	0002	COMMON PGHIDS\$ 'PROGRAM OR FORM ID
001A	0002	COMMON MOLENGTH() 'DAYS IN THE MONTH
001A	0002	COMMON USERS()
001A	0002	' *****END OF COMMON DEFINITION*****
001A	0002	
001A	0002	
001A	0002	REM \$INCLUDE: 'RACDEF.MOD' REM INCLUDE THE DEFAULT DEFINITIONS
001A	0002	*****
001A	0002	* NAME: RACP01.DEF DEFAULT DEFINITIONS *
001A	0002	* Date: 28 Feb 84 Written by: Floyd Cole *
001A	0002	*****
001A	0002	' Variables used in common that have a default value on start*up
001A	0002	' will be held in this file. It is an included file so it cannot
001A	0002	' be run in a stand*alone mode. In normal operation, this file
001A	0002	' should be 'included' in the main program only (RACP10.BAS).
001A	0002	'
001A	0002	' This program segment may be modified, but all files containing
001A	0002	' an include for this segment must be re*compiled in order to
001A	0002	' affect the changes made here.
001A	0002	'
001A	0002	
001A	0002	' *****START OF DEFAULT DEFINITION*****
001A	0002	FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
0049	1100	BACK = 1 'Background Color = Light Blue
0050	1100	BORD = 4 'BORDER = RED
0077	1102	HIDE = 4 'ALTERNATE COLOR = RED
005E	1102	EFORE= 14 'ERROR FOREGROUND DISPLAY
0065	1102	EBACK= 0 'ERROR BACKGROUND DISPLAY
006C	1102	BELL\$ = CHR\$(7) 'Sound the bell
0078	1102	

RACP940
SHORT FORM

PAGE 4
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.30

Offset	Data	Source Line
0078	1102	MOLENGTH(1) = 31 'JAN
007F	1102	MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
0086	1102	MOLENGTH(3) = 31 'MAR
008D	1102	MOLENGTH(4) = 30 'APR
0094	1102	MOLENGTH(5) = 31 'MAY
009B	1102	MOLENGTH(6) = 30 'JUN
00A2	1102	MOLENGTH(7) = 31 'JUL
00A9	1102	MOLENGTH(8) = 31 'AUG
00B0	1102	MOLENGTH(9) = 30 'SEP
00B7	1102	MOLENGTH(10) = 31 'OCT
00BE	1102	MOLENGTH(11) = 30 'NOV
00C5	1102	MOLENGTH(12) = 31 'DEC
00CC	1102	
00CC	1102	DATEERR\$(0) = " "
00D5	1102	DATEERR\$(1) = "INVALID MONTH"
00DE	1102	DATEERR\$(2) = "INVALID DAY "
00E7	1102	DATEERR\$(3) = "DAY TOO LARGE FOR MONTH CODED"
00F0	1102	
00F0	1102	MAXLENGTH = 80 'MAXIMUM LENGTH OF OUTPUT RECORD
00F7	1104	PADS = " " 'PAD CHARACTER FOR SHORT RECORDS
0100	1108	
0100	1108	' *****END OF DEFAULT DEFINITION*****
0100	1108	
0100	1108	
0100	1108	KEY OFF
0106	1108	
0106	1108	REM *****
0106	1108	REM THE FOLLOWING VARIABLES ARE UNIQUE TO EACH PROGRAM AND MUST
0106	1108	REM BE CHANGED.
0106	1108	REM *****
0106	1108	PGMTITL\$ = "SHORT FORM"
010F	110C	
010F	110C	PGMIDS = "940" 'VALUE RECEIVED FROM THE SCANNER
0118	110C	'IN HEADER VARIABLE 'PROGRAMS'
0118	110C	
0118	110C	DATFIL\$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
0121	1110	
0121	1110	REM LENGTH OF STRING RECEIVED FROM THE OMR....
0121	1110	HEADER = 21
0128	1112	RESPONSE= 231
012F	1114	RECORDLENGTH = HEADER + RESPONSE
013A	1116	
013A	1116	N.PROC = 16 ' NUMBER OF PROCEDURES FOR THIS FORM
0141	1118	
0141	1118	REM *****
0141	1118	
0141	1118	BTIMES=TIMES 'SCAN START TIME
014A	111C	
014A	111C	
014A	111C	REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
014A	111C	REM CLINIC #1
014A	111C	CLINIC1.PFX\$(0)=" "
0153	111C	CLINIC1.PFX\$(1)="B"
015C	111C	CLINIC1.PFX\$(2)="D"

RACP940
SHORT FORM

PAGE 5
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```
0165 111C CLINIC1.PFX$(3)="F"
016E 111C CLINIC1.PFX$(4)="G"
0177 111C CLINIC1.PFX$(5)="S"
0180 111C
0180 111C REM *** PROVIDER TIME TABLE ***
0180 111C PROVIDER.TIMES(0)="000" 'NO TIME
0189 111C PROVIDER.TIMES(1)="002" ' 2 MINUTES
0192 111C PROVIDER.TIMES(2)="005" ' 5 MINUTES
0198 111C PROVIDER.TIMES(3)="010" '10 MINUTES
01A4 111C PROVIDER.TIMES(4)="015" '15 MINUTES
01AD 111C PROVIDER.TIMES(5)="020" '20 MINUTES
01B6 111C PROVIDER.TIMES(6)="030" '30 MINUTES
01BF 111C PROVIDER.TIMES(7)="040" '40 MINUTES
01C8 111C PROVIDER.TIMES(8)="050" '50 MINUTES
01D1 111C PROVIDER.TIMES(9)="060" '60 MINUTES
01DA 111C
01DA 111C REM *** ENCOUNTER FORM PROCEDURE TABLE ***
01DA 111C PROCED$(0)=" " : PROCED$(07) = "90037" : PROCED$(14)= "87060"
01F5 111C PROCED$(1)="59420" : PROCED$(08) = "99096" : PROCED$(15)= "36415"
0210 111C PROCED$(2)="93786" : PROCED$(09) = "99094" : PROCED$(16)= "99097"
0228 111C PROCED$(3)="90657" : PROCED$(10) = "90700"
0230 111C PROCED$(4)="99091" : PROCED$(11) = "86581"
024F 111C PROCED$(5)="99095" : PROCED$(12) = "86582"
0261 111C PROCED$(6)="99092" : PROCED$(13) = "90601"
0273 111C
0273 111C REM INCLUDE: 'UCABAMC.OPT' INCLUDE OUTPATIENT UCA TABLE
0273 111C
0273 111C REM $PAGE
```

RACP940
SHORT FORM

PAGE 6
07-06-87
15:07:19

Offset Date Source Line

IBM Personal Computer BASIC Compiler V1.00

```
0273 111C      GOSUB 1000      'MAKE SURE THEY ARE LOGGED ON
0278 111C      CLS
027C 111C      GOSUB 7000      'PRINT SCREEN HEADING
0281 111C
0281 111C  REM *****
0281 111C  REM ****          OPEN FILE TO CONTAIN SCANNED DATA          ****
0281 111C  REM *****
0281 111C  REM
0281 111C      OPEN DATFIL$ FOR APPEND AS #1
0293 111C
0293 111C  REM *****
0293 111C  REM ****          CLEAR AND DISPLAY PROGRAM SCREEN          ****
0293 111C  REM *****
0293 111C      LPRINT CHR$(15);
029E 111C      WIDTH "LPT1:",160
02A8 111C      PAGE = 0 : GOSUB 7100      'LINE PRINTER HEADING
02B4 111E      COLOR 14
02B8 111E      LOCATE 11,26 : PRINT "SHORT FORM "
02D0 111E      COLOR FORE,BACK,BORD
02E6 111E
02E6 111E  REM *****
02E6 111E  REM ****          COMMUNICATIONS SETUP          ****
02E6 111E  REM *****
02E6 111E  REM PROTOCOL
02E6 111E      GOSUB 9001
02EB 111E      IF ERRMSG$ > " " THEN LPRINT ERRMSG$ : GOTO 30000
0305 1122
0305 1122  REM START SCANNER (S1)
0305 1122      CNTRLOPT =1 :GOSUB 9010
0311 1124      IF ERRMSG$ > " " THEN LPRINT ERRMSG$ : GOTO 30000
0328 1124
0328 1124      LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
0340 1124      READYTYPE=3          'FIRST TIME IN.. SCANNER IS STARTED..
0347 1126
0347 1126  REM *****
0347 1126  REM ****          SET SCAN SHEET CALL          ****
0347 1126  REM *****
0347 1126  REM
0347 1126
0347 1126  10 REM - RETURN POINT TO READ NEXT SHEET
0348 1126
0348 1126      AS=INKEY$
0351 112A      IF AS=CHR$(27) THEN GOTO 25000
0367 112A
0367 112A      GOSUB 9020          'SCAN SUBROUTINE - GET A RECORD
036C 112A      IF MID$(ERRSTAT$,14,3)="415" THEN GOTO 25000
0388 112E
0388 112E      TEXT$=""          'CLEAR THE INPUT AREA
0391 112E      GOSUB 8000          'DECODE HEADER
0396 112E      GOSUB 8050          'CHECK FOR END OF JOB/END OF BATCH
0398 112E      GOSUB 8200          'DECODE THE RESPONSE POSITIONS
03A0 112E      LITHOS = MID$(TEXT$,22,8)
0382 1132      GOSUB 8070          'CHECK FOR SCANNER ERRORS
0387 1132      GOSUB 8100          'PRINT THE DATA ON THE SCREEN
```

RACP940
SHORT FORM

PAGE 7
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```
038C 1132
038C 1132 REM $INCLUDE: 'RACP940.M01' INCLUDE THE SHORT FORM REFORMAT/EDIT MOD
038C 1132 REM *****
038C 1132 REM **** AMBULATORY CARE INFORMATION SYSTEM 30 JUN 87 ****
038C 1132 REM **** D R BOLLING ****
038C 1132 REM **** MODULE NAME : RACP940.M01 ****
038C 1132 REM **** SCANNER PROGRAM # : 940-SHORT FORM ****
038C 1132 REM ****
038C 1132 REM **** PURPOSE : REFORMAT/EDIT THE ENCOUNTER ****
038C 1132 REM **** SHORT FORM OMR RECORD. ****
038C 1132 REM ****
038C 1132 REM **** MODIFIED 30 JUN 87 TO ALLOW SPEC PRE CODES TO BE ****
038C 1132 REM **** BE BUBBLED FOR PROCEDURE. ****
038C 1132 REM *****
038C 1132 REM **** RESERVED LINE NUMBERS 100-199 ****
038C 1132 REM *****
038C 1132
038C 1132 N.ERR =0 'COUNTS THE NUMBER OF ERRORS
03C3 1134
03C3 1134 REM *** LITHO CODE DONE IN BAS PROGRAM ***
03C3 1134
03C3 1134 REM *** CLINIC ID ***
03C3 1134 100 CK.X=VAL(MID$(TEXT$,30,1))
03D9 1136 CK.CODS=MID$(TEXT$,31,3)
03EB 113A REM GOSUB 5500 'VALIDATE OUTP UCA CODE
03EB 113A REM IF RT.5500 = 0 THEN GOTO 101
03EB 113A REM N.ERR = N.ERR + 1
03EB 113A REM ED.MSG$(N.ERR)="#1-2 INVALID CLINIC CODE"
03EB 113A
03EB 113A CL1.CODS=CLINIC1.PFX$(CK.X) + CK.CODS
0403 113E
0403 113E REM *** VISIT DATE ***
0403 113E 104 CMS=MID$(DATE$,1,2) 'CURRENT MONTH
0415 1142 YRS=MID$(DATE$,9,2) 'CURRENT YEAR
0427 1146 XS=MID$(TEXT$,34,4) 'MONTH AND DAY FROM FORM
0439 114A IF LEFT$(XS,2)<=CMS THEN 105 'OK, USE THIS YEAR
044E 114A YRS=RIGHT$(STR$(VAL(YRS)-1),2) 'USE LAST YEAR
045E 114A 105 VDATES = YRS + XS
047C 114E D1$=MID$(XS,3,1)
048E 1152 D2$=MID$(XS,4,1)
04A0 1156 IF D1$=" " AND D2$<>" " THEN RT.5000=2 : GOTO 106
04CE 1158 IF D1$<>" " AND D2$=" " THEN RT.5000=2 : GOTO 106
04FC 1158 'EDIT VISIT DATE
04FC 1158 CK.5000$=VDATES
0505 115C GOSUB 5000 'DATE CHECK
050A 115C CK.5010$=VDATES
0513 1160 GOSUB 5010 'NUMERIC STRING CHECK
0518 1160 IF RT.5000 = 0 AND RT.5010 = 0 THEN GOTO 107
053C 1162 106 N.ERR=N.ERR+1
054 1162 ED.MSG$(N.ERR)="TODAYS DATE " + DATEERR$(RT.5000)
056A 1162
056A 1162 REM *** PROVIDER ID (PREFIX + NUM) ***
056A 1162 107 PROV1.PFX$ = MID$(TEXT$,38,1)
057C 1166 PROV1.NUM$ = MID$(TEXT$,39,4)
```

RACP940
SHORT FORM

PAGE 8
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
058E	116A	
058E	116A	
058E	116A	SSN.OFFSET = 43
0595	116C	FMP.OFFSET = 52
059C	116E	TIM.OFFSET = 54
05A3	1170	UNL.OFFSET = 55
05AA	1172	PRC.OFFSET = 60
05B1	1174	SPE.OFFSET = 76
0588	1176	TOT.OFFSET = 42
05BF	1178	POINTER = 0
05C6	117A	
05C6	117A	FOR I940 = 1 TO 5
05CD	117A	SPE.COD\$(I940)=" " 'INITIALIZE
05E1	117C	NEXT I940
05F0	117C	
05F0	117C	REM LOOP THROUGH THE FOLLOWING CODE 5 TIMES (1 FOR EACH GROUP)
05F0	117C	FOR I411 = 1 TO 5
05F7	117C	REM PATIENT ID (SSN+FMP)
05F7	117C	110 SSN\$=MID\$(TEXT\$,SSN.OFFSET+POINTER,9)
060E	1180	FMP\$=MID\$(TEXT\$,FMP.OFFSET+POINTER,2)
0625	1184	
0625	1184	PTID\$(I411) = SSN\$+FMP\$
063E	1186	IF I411=1 THEN GOTO 112
064D	1186	IF PTID\$(I411)=STRING\$(11," ") THEN GOTO 130
0673	1186	
0673	1186	REM *** PROVIDER TIME ***
0673	1186	112 X=VAL(MID\$(TEXT\$,TIM.OFFSET+POINTER,1))
068E	1188	PTIM\$(I411)=PROVIDER.TIME\$(X)
06A8	1188	IF PTIM\$(I411)<>"000" THEN 114
06C3	1188	N.ERR=N.ERR+1
06CB	1188	ED.MSG\$(N.ERR)="NO TIME CODED FOR PART "+STR\$(I411)
06E8	1188	
06E8	1188	REM *** UNLISTED PROCEDURE ***
06E8	1188	114 ADDP\$=MID\$(TEXT\$,UNL.OFFSET+POINTER,5)
06FF	118C	
06FF	118C	REM *** PROCEDURE CODES ***
06FF	118C	120 X\$ = MID\$(TEXT\$,PRC.OFFSET+POINTER,N.PROC)
0717	118C	GOSUB 6000 'DECODE THE X\$ STRING
071C	118C	IF TOT = 0 THEN GOTO 122
072B	118E	TOT.PROC(I411) = TOT
073A	118E	FOR I411.A= 1 TO TOT
0747	1190	PTR=VAL(HOLD\$(I411.A))
075D	1194	GROUP\$(I411,I411.A)=PROC\$(PTR)
077F	1194	NEXT I411.A
0790	1194	
0790	1194	122 IF ADDP\$=" " THEN 126
079E	1194	TOT=TOT + 1
07A6	1194	GROUP\$(I411,TOT)=ADDP\$ 'ADD UNLISTED PROC
07BF	1194	TOT.PROC(I411)=TOT
07CE	1194	
07CE	1194	REM *** SPECIFIC PREASSIGNED CODES ***
07CE	1194	126 INP.STO\$=MID\$(TEXT\$,SPE.OFFSET+POINTER,9)
07E5	1198	GOSUB 5700 'CONVERT ARRAY

```

Offset  Data  Source Line
-----
07EA 1198      SPE.COD$(I411)=BUF.STOS      'UP TO 9 TWO DIGIT CODES
07FE 119C
07FE 119C
07FE 119C      IF TOT=0 AND SPE.COD$(I411)=" THEN 128 ELSE 130
0830 119C
0830 119C      128  N.ERR=N.ERR+1
0838 119C      ED.MSG$(N.ERR)="NO PROCEDURE OR SPEC PRE CL CODE IN PART "+STR$(I411)
0855 119C
0855 119C      130  POINTER = POINTER + TOT.OFFSET
0860 119C      NEXT I411
0872 119C      199  REM
0873 119C
0873 119C      REM -----END OF MODULE  RACH940.MO1-----
0873 119C
0873 119C      IF N.ERR = 0 THEN GOTO 997
0882 119C      LPRINT "LITHO # ";LITHOS;" ... ERRORS"
0894 119C      FOR I997 = 1 TO N.ERR
08A1 119E      LPRINT USING "### ";I997;
08AD 11A0      LPRINT "=> ";ED.MSG$(I997)
08C3 11A0      NEXT I997
08D4 11A0      LN.COUNT = LN.COUNT + N.ERR + 1
08E0 11A2      CNTRLOPT = 6
08E7 11A2      GOSUB 9010      'REJECT THE FORM
08EC 11A2      GOTO 998      'BYPASS THE DISK WRITER....
08F0 11A2
08F0 11A2      997  REM $INCLUDE: 'RACH940.MO2'      REM INCLUDE THE BASE ENCOUNTER FORM DISK WRITER
08F1 11A2      REM *****
08F1 11A2      REM ****      AMBULATORY CARE INFORMATION SYSTEM      19 MAR 87      ****
08F1 11A2      REM ****      D R BOLLING      ****
08F1 11A2      REM ****      MODULE NAME      :      RACH940.MO2      ****
08F1 11A2      REM ****      SCANNER PROGRAM #      :      940-SHORT FORM      ****
08F1 11A2      REM ****
08F1 11A2      REM ****      PURPOSE      :      CREATE AND WRITE THE DISK      ****
08F1 11A2      REM ****      RECORD FOR INPUT TO FOCUS      ****
08F1 11A2      REM ****
08F1 11A2      REM ****      PROGRAM ADDS PREFIX TO LITHO FOR EACH PATIENT      ****
08F1 11A2      REM *****
08F1 11A2      REM ****      RESERVED LINE NUMBERS 200-299      ****
08F1 11A2      REM *****
08F1 11A2
08F1 11A2      REM      BUILD THE OUTPUT RECORD
08F1 11A2
08F1 11A2      GOSUB 270      'BUILD THE RECORD KEY
08F6 11A2
08F6 11A2      REM *****
08F6 11A2      REM ****      RECORD TYPE "1" - RECKEY PLUS TRANSACTION 1 DATA      ****
08F6 11A2      REM *****
08F6 11A2      VCNT$ = "0"      'DEFAULT COUNT IS 0 FOR THIS FORM
08FF 11A6      FRM$ = "94"      'FORM NUMBER
0908 11AA
0908 11AA      FOR I412 = 1 TO 5
090F 11AA      IF PTID$(I412)=STRING$(11," ") THEN GOTO 210
0935 11AC      RECOU$ = "9401" + RECKEY$ + RIGHT$(STR$(I412),1) + RIGHT$(LITHOS,7)

```

RACP940
SHORT FORM

PAGE 10
07-06-87
15:07:19

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```
0965 1184      MIDS(RECOUT$,20,11)=PTIDS(I412)
0970 1184      RECOUT$ =RECOUT$ + FRMNS + VCNTS +PTIMS(I412) + STRINGS(30," ")
0988 1184      GOSUB 280
098D 1184      PRINT #1,RECOUT$
09C8 1184  210  NEXT I412
09DA 1184
09DA 1184  REM *****
09DA 1184  REM ****  RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE  ****
09DA 1184  REM *****
09DA 1184  REM RECOUT$ ="9402"+RECKEY$ 'TRANSACTION IDENTIFIER
09DA 1184
09DA 1184      FOR I412 = 1 TO 5
09E1 1184          IF PTIDS(I412)=STRINGS(11," ") THEN GOTO 250
0A07 1184          FOR I412.A = 1 TO TOT.PROC(I412)
0A18 1186      RECOUT$ = "9402" + RECKEY$ + RIGHTS(STR$(I412),1) + RIGHTS(LITHOS,7)
0A48 1186          MIDS(RECOUT$,20,11)=PTIDS(I412)
0A63 1186      RECOUT$ =RECOUT$ + FRMNS + "1" + GROUP$(I412,I412.A)
0A8F 1188      GOSUB 280
0A94 1188      PRINT #1,RECOUT$
0A9F 1188      NEXT I412.A
0AB3 1188  250  NEXT I412
0AC5 1188
0AC5 1188  REM *****
0AC5 1188  REM ****  RECORD TYPE "3" - RECKEY PLUS SPE PRE CLINIC CODES  ****
0AC5 1188  REM *****
0AC5 1188      FOR I412 = 1 TO 5
0ACC 1188          IF LEN(SPE.COD$(I412))=0 THEN 260
0AE2 1188          RPOINT = 1
0AE9 118A
0AE9 118A  252  RECOD3$=RIGHT$(MIDS(SPE.COD$(I412),RPOINT,2),1)
0B08 118E      IF RECOD3$=" " THEN 254
0B19 118E
0B19 118E      RECOUT$ = "9403" + RECKEY$ + RIGHTS(STR$(I412),1) + RIGHTS(LITHOS,7)
0B49 118E          MIDS(RECOUT$,20,11)=PTIDS(I412)
0B61 118E      RECOUT$ =RECOUT$ + FRMNS + RECOD3$
0B74 118E      GOSUB 280
0B79 118E      PRINT #1,RECOUT$
0B84 118E
0B84 118E  254  RPOINT = RPOINT + 2
0B8D 118E      IF RPOINT < LEN(SPE.COD$(I412)) THEN 252
0BA5 118E
0BA5 118E  260  NEXT I412
0BB7 118E
0BB7 118E      REM END OF TYPE 3 RECORDS
0BB7 118E
0BB7 118E      GOTO 299
0BB8 118E
0BB8 118E  REM *****
0BB8 118E  REM ****  SUBROUTINE 270 - BUILD THE RECORD KEY  ****
0BB8 118E  REM *****
0BB8 118E  270  RECKEY$=""
0BC4 118E
0BC4 118E  REM ***  CLINIC ID (PREFIX + CODE) ***
0BC4 118E      RECKEY$= CL1.COD$
```

RACP940
SHORT FORM

PAGE 11
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
08CD	11BE	
08CD	11BE	REM *** VISIT DATE ***
08CD	11BE	RECKEYS=RECKEYS+ VDATES
08D9	11BE	
08D9	11BE	REM *** PROVIDER ID (PREFIX + NUM) ***
08D9	11BE	RECKEYS = RECKEYS + PROV1.PFX\$ + PROV1.NUM\$
08EC	11BE	
08EC	11BE	REM *** PATIENT ID (SSN+FMP) ***
08EC	11BE	RECKEYS = RECKEYS + STRINGS(11," ")
08FE	11BE	
08FE	11BE	REM *** LITHO CODE ***
08FE	11BE	REM DO LITHO IN LOOP ABOVE
08FE	11BE	REM RECKEYS = RECKEYS + LITHOS
08FE	11BE	
08FE	11BE	RETURN
0C01	11BE	
0C01	11BE	REM *****
0C01	11BE	REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
0C01	11BE	REM *****
0C01	11BE	280 PAD=MAXLENGTH - LEN(RECOUT\$) 'FIND OUT HOW SHORT THE RECORD IS
0C0F	11C0	RECOUT\$ = RECOUT\$ + STRINGS(PAD,PAD\$) 'PAD THE RECORD WITH FILL CHAR
0C22	11C0	RETURN
0C25	11C0	
0C25	11C0	
0C25	11C0	299 REM
0C26	11C0	
0C26	11C0	REM -----END OF MODULE RACP940.MO2-----
0C26	11C0	
0C26	11C0	998 REM CONTINUE
0C27	11C0	
0C27	11C0	999 READTYPE = 2
0C2E	11C0	IF LN.COUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
0C3E	11C0	GOTO 10
0C42	11C0	
0C42	11C0	REM END OF SCAN/DECODE/WRITE LOOP =====
0C42	11C0	
0C42	11C0	1000 REM \$INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
0C43	11C0	REM *****
0C43	11C0	REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
0C43	11C0	REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
0C43	11C0	REM *****
0C43	11C0	REM PATIENT OMR INPUT PROGRAM *
0C43	11C0	REM *
0C43	11C0	REM This program verifies user is logged on properly. If there is no *
0C43	11C0	REM valid user logged on at the time of execution, this subroutine will*
0C43	11C0	REM chain to the logon program RACP05, otherwise a return is issued. *
0C43	11C0	REM *****
0C43	11C0	REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
0C43	11C0	REM *****
0C43	11C0	1001 OPEN "I",1,"RACLOG.DAT"
0C55	11C0	IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
0C63	11C0	INPUT #1,USERS(1),DT\$,TMS\$,PID\$
0C74	11CC	IF USERS(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
0C92	11CC	IF USERS(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST

RACP940
SHORT FORM

PAGE 12
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
OCA0	11CC	CLOSE 1
OCA7	11CC	SCREEN 0,1,0,0
OCB0	11CC	COLOR FORE,BACK,BORD
OCD3	11CC	CLS
OCD7	11CC	RETURN
OCDA	11CC	
OCDA	11CC	1002 CLOSE
OCDE	11CC	CHAIN "RACP05"
OCE5	11CC	'*****END OF LOGON VERIFY SUBROUTINE 1000*****'
OCE5	11CC	
OCE5	11CC	2000 REM \$INCLUDE: 'RACS2000.SUB' INCLUDE THE REPLY/DELAY SUB
OCE6	11CC	REM *****
OCE6	11CC	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
OCE6	11CC	REM **** SKIP COLE ****
OCE6	11CC	REM **** SUBROUTINE NAME : RACS2000.SUB ****
OCE6	11CC	REM **** SCANNER PROGRAM # : ALL ****
OCE6	11CC	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
OCE6	11CC	REM **** SERVERS AS A WAIT AND REPLY ****
OCE6	11CC	REM **** ENTRY MODULE ****
OCE6	11CC	REM **** INPUT : SINGLE KEYBOARD ENTRY ****
OCE6	11CC	REM ****
OCE6	11CC	REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
OCE6	11CC	REM ****
OCE6	11CC	REM **** RESERVED LINE ****
OCE6	11CC	REM **** NUMBERS : 2001-2010 ****
OCE6	11CC	REM *****
OCE6	11CC	2001 REM REPLY FUNCTION
OCE7	11CC	2002 REPLY\$=INKEY\$: IF REPLY\$="" THEN 2002
OCF8	11D0	REPLY=ASC(REPLY\$)
OD05	11D2	IF REPLY > 90 THEN REPLY\$=CHR\$(REPLY XOR 32) 'CONVERT TO CAPS
OD20	11D2	IF REPLY\$ < "A" OR REPLY\$ > "Z" THEN REPLY\$="?"
OD4C	11D2	RETURN
OD4F	11D2	
OD4F	11D2	5000 REM \$INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB
OD50	11D2	REM *****
OD50	11D2	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
OD50	11D2	REM **** SKIP COLE ****
OD50	11D2	REM **** SUBROUTINE NAME : RXXS5000.SUB ****
OD50	11D2	REM **** SCANNER PROGRAM # : ALL ****
OD50	11D2	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
OD50	11D2	REM **** PERFORMS A DATE EDIT ****
OD50	11D2	REM ****
OD50	11D2	REM **** INPUT : DATE TO BE CHECKED MUST BE ****
OD50	11D2	REM **** IN THE VARIABLE NAMED ****
OD50	11D2	REM **** 'CK.5000\$' ****
OD50	11D2	REM **** IN THE FORMAT "YYMMDD" ****
OD50	11D2	REM ****
OD50	11D2	REM **** OUTPUT : 'RT.5000' IS THE RETURN CODE ****
OD50	11D2	REM **** VARIABLE. IF THIS VARIABLE ****
OD50	11D2	REM **** CONTAINS ANY NUMBER OTHER ****
OD50	11D2	REM **** THAN 0, AN ERROR WAS FOUND ****
OD50	11D2	REM **** IN THE DATE. ****
OD50	11D2	REM ****
OD50	11D2	REM **** RESERVED LINE ****

Offset	Data	Source Line
0050	11D2	REM **** NUMBERS : 5001-5009 ****
0050	11D2	REM ****
0050	11D2	RT.5000 = 0
0057	11D2	CKYEAR = VAL(LEFT\$(CK.5000\$,2)) 'YEAR NUMERIC VALUE
006A	11D4	CKMONTH = VAL(MID\$(CK.5000\$,3,2)) 'MONTH NUMERIC VALUE
0080	11D6	CKDAY = VAL(RIGHT\$(CK.5000\$,2)) 'DAY NUMERIC VALUE
0093	11D8	
0093	11D8	IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
00A9	11D8	IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
00BF	11D8	IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
00D5	11D8	IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
00EB	11D8	
00EB	11D8	REM LEAP YEAR CHECK
00EB	11D8	MOLENGTH(2) = 28
00F2	11D8	IF CKMONTH<> 2 THEN GOTO 5005 'MUST BE FEBRUARY
0E01	11D8	IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005 'MUST BE A LEAP YEAR
0E16	11D8	MOLENGTH(2) = 29
0E1D	11D8	
0E1D	11D8	5005 IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
0E3C	11D8	
0E3C	11D8	5009 RETURN
0E3F	11D8	
0E3F	11D8	REM -----END OF SUBROUTINE 5000 -----
0E3F	11D8	
0E3F	11D8	5010 REM \$INCLUDE: 'RACS5010.SUB' INCLUDE THE NUMERIC STRING EDITOR
0E40	11D8	REM *****
0E40	11D8	REM **** AMBULATORY CARE DATA BASE 1 MAY 85 ****
0E40	11D8	REM **** SKIP COLE ****
0E40	11D8	REM **** SUBROUTINE NAME : RXXS5010.SUB ****
0E40	11D8	REM **** SCANNER PROGRAM # : ALL ****
0E40	11D8	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0E40	11D8	PERFORMS A NUMERIC STRING ****
0E40	11D8	EDIT. ****
0E40	11D8	REM ****
0E40	11D8	REM **** INPUT : STRING TO BE EDITED IS IN ****
0E40	11D8	THE VARIABLE NAMED ****
0E40	11D8	'CK.5010\$' ****
0E40	11D8	REM ****
0E40	11D8	REM **** OUTPUT : 'RT.5010' IS THE RETURN CODE ****
0E40	11D8	VARIABLE. IF THIS VARIABLE ****
0E40	11D8	CONTAINS ANY NUMBER OTHER ****
0E40	11D8	THAN 0, AN ERROR WAS FOUND ****
0E40	11D8	IN THE STRING. ****
0E40	11D8	REM ****
0E40	11D8	REM **** RESERVED LINE ****
0E40	11D8	REM **** NUMBERS : 5011-5019 ****
0E40	11D8	REM *****
0E47	11D8	RT.5010 = 0
0E47	11D8	
0E47	11D8	FOR I5010 = 1 TO LEN(CK.5010\$)
0E57	11DA	J5010= ASC(MID\$(CK.5010\$,I5010,1))
0E6B	11DE	IF J5010 < 48 OR J5010 > 57 THEN RT.5010 = RT.5010 + 1
0E93	11DE	NEXT I5010
0EA4	11DE	

PAGE 14
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.00

Address	Disassembly	Comments
0EA4 11DE	RETURN	
0EA7 11DE	REM ----- END OF SUBROUTINE 5010 -----	
0EA7 11DE	5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTPATIENT UCA CHECK SUB	
0EA8 11DE	5700 REM \$INCLUDE: 'RACS5700.SUB' INCLUDE THE ARRAY CONVERTER SUB	
0EA9 11DE	REM *****	
0EA9 11DE	REM **** AMBULATORY CARE DATA BASE 29 JUL 85 ****	
0EA9 11DE	REM **** D R BOLLING ****	
0EA9 11DE	REM **** SUBROUTINE NAME : RXXS5700.SUB ****	
0EA9 11DE	REM **** SCANNER PROGRAM # : ALL ****	
0EA9 11DE	REM **** FUNCTION : THIS SUBROUTINE MODULE ****	
0EA9 11DE	REM **** CONVERTS A BINARY ARRAY INTO ****	
0EA9 11DE	REM **** TWO CHAR CODES. ****	
0EA9 11DE	REM ****	
0EA9 11DE	REM **** INPUT : INP.STO\$ AS STRING ****	
0EA9 11DE	REM ****	
0EA9 11DE	REM ****	
0EA9 11DE	REM **** OUTPUT : BUF.STO\$ AS STRING. ****	
0EA9 11DE	REM ****	
0EA9 11DE	REM ****	
0EA9 11DE	REM **** RESERVED LINE ****	
0EA9 11DE	REM **** NUMBERS : 5710-5730 ****	
0EA9 11DE	REM *****	
0EA9 11DE		
0EA9 11DE	BUF.STO\$=""	
0EB2 11DE	N.STO=1	
0EB9 11E0	5710 X.STO=INSTR(N.STO,INP.STO\$,"1")	
0ECB 11E2	IF X.STO=0 THEN GOTO 5720 'THATS ALL	
0EDA 11E2	N.STO = X.STO + 1 'NEXT STARTING POINT	
0EE2 11E2	X.STO = X.STO + 100 'PAD WITH LEADING ZERO	
0EEC 11E2	BUF.STO\$ = BUF.STO\$ + RIGHT\$(STR\$(X.STO),2)	
0F02 11E2	IF N.STO <= LEN(INP.STO\$) THEN GOTO 5710	
0F15 11E2		
0F15 11E2	5720 RETURN	
0F18 11E2		
0F18 11E2	REM ----- END OF SUBROUTINE 5700 -----	
0F18 11E2		
0F18 11E2	6000 REM \$INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE ROUTINE	
0F19 11E2	REM *****	
0F19 11E2	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****	
0F19 11E2	REM **** SKIP COLE ****	
0F19 11E2	REM **** SUBROUTINE NAME : RXXS6000.SUB ****	
0F19 11E2	REM **** SCANNER PROGRAM # : ALL ****	
0F19 11E2	REM **** FUNCTION : THIS SUBROUTINE MODULE ****	
0F19 11E2	REM **** PERFORMS INSTRING SEARCH ****	
0F19 11E2	REM ****	
0F19 11E2	REM **** INPUT : STRING TO BE SEARCHED MUST ****	
0F19 11E2	REM **** BE IN THE VARIABLE NAMED : ****	
0F19 11E2	REM **** 'XS' ****	
0F19 11E2	REM ****	
0F19 11E2	REM **** OUTPUT : 'TOT' = TOTAL NUMBER OF ****	
0F19 11E2	REM **** HITS IN THE DESTRING ****	
0F19 11E2	REM **** 'HOLD\$()' IS THE ARRAY ****	
0F19 11E2	REM **** CONTAINING THE NUMERIC ****	
0F19 11E2	REM **** VALUE OF THE HIT POSITIONS ****	

RACP940
SHORT FORM

PAGE 15
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
0F19	11E2	REM ****
0F19	11E2	REM *****
0F19	11E2	REM **** RESERVED LINE ****
0F19	11E2	REM **** NUMBERS : 6001-6009 ****
0F19	11E2	REM *****
0F19	11E2	6001 PTR = INSTR(X\$, "1")
0F27	11E2	TOT = 0
0F2E	11E2	WHILE PTR > 0
0F39	11E2	TOT=TOT+1
0F41	11E2	HOLD\$(TOT) = RIGHT\$(STR\$(PTR), 2)
0F63	11E2	PTR=PTR+1
0F68	11E2	PTR = INSTR(PTR, X\$, "1")
0F7D	11E2	WEND
0F81	11E2	RETURN
0F84	11E2	
0F84	11E2	REM -----END OF SUBROUTINE RXXS6000.SUB-----
0F84	11E2	
0F84	11E2	
0F84	11E2	7000 REM \$INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
0F85	11E2	REM *****
0F85	11E2	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0F85	11E2	REM **** SKIP COLE ****
0F85	11E2	REM **** SUBROUTINE NAME : RACS7000.SUB ****
0F85	11E2	REM **** SCANNER PROGRAM # : ALL ****
0F85	11E2	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0F85	11E2	REM **** PRINTS THE STANDARD SCREEN ****
0F85	11E2	REM **** HEADING. ****
0F85	11E2	REM **** INPUT : COMMON VARIABLE USERS(2) ****
0F85	11E2	REM **** SYSTEM DATE ****
0F85	11E2	REM ****
0F85	11E2	REM **** OUTPUT : SCREEN HEADING ****
0F85	11E2	REM ****
0F85	11E2	REM **** RESERVED LINE ****
0F85	11E2	REM **** NUMBERS : 7001-7010 ****
0F85	11E2	REM *****
0F85	11E2	
0F85	11E2	7001 LOCATE 1,1
0F8F	11E2	PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
0F97	11E2	LOCATE 1,65
0FA4	11E2	PRINT DATES;
0FAC	11E2	LOCATE 2,1
0FB9	11E2	PRINT "USER : ";USERS(1)
0FC6	11E2	RETURN
0FC9	11E2	7100 REM \$INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
0FCA	11E2	REM *****
0FCA	11E2	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0FCA	11E2	REM **** SKIP COLE ****
0FCA	11E2	REM **** SUBROUTINE NAME : RXXS7100.SUB ****
0FCA	11E2	REM **** SCANNER PROGRAM # : ALL ****
0FCA	11E2	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0FCA	11E2	REM **** PRINTS THE STANDARD HEADING ****
0FCA	11E2	REM **** ON THE PRINTER. ****
0FCA	11E2	REM **** INPUT : DATE,PAGE,PGMID\$,PGMTITL\$ ****
0FCA	11E2	REM ****

```

0FCA 11E2 REM **** OUTPUT : PRINTER HEADING, LN.COUNT ****
0FCA 11E2 REM ****
0FCA 11E2 REM **** RESERVED LINE ****
0FCA 11E2 REM **** NUMBERS : 7101-7110 ****
0FCA 11E2 REM ****
0FCA 11E2
0FCA 11E2 7101 IF PAGE > 0 THEN LPRINT CHR$(12);
0FE0 11E2 LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITL$;
0FED 11E2 LPRINT TAB(70);DATE$
1000 11E2 PAGE=PAGE+1
1008 11E2 LPRINT "PROGRAM ";PGMIDS;TAB(70);"PAGE";
1025 11E2 LPRINT USING "###";PAGE
1031 11E2 LPRINT
1039 11E2 LN.COUNT=3
1040 11E2 RETURN
1043 11E2
1043 11E2 8000 REM $INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
1044 11E2 REM *****
1044 11E2 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ***
1044 11E2 REM **** SKIP COLE ***
1044 11E2 REM **** SUBROUTINE NAME : RXXS8000.SUB ***
1044 11E2 REM **** SCANNER PROGRAM # : ALL ***
1044 11E2 REM **** FUNCTION : THIS SUBROUTINE MODULE ***
1044 11E2 REM **** IS A GROUPING THAT PERFORMS ***
1044 11E2 REM **** VARIOUS DECODING FUNCTIONS ***
1044 11E2 REM **** ON THE SCANNER DATA ***
1044 11E2 REM ****
1044 11E2 REM **** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
1044 11E2 REM **** 8050 - CHECK FOR END OF JOB ***
1044 11E2 REM **** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
1044 11E2 REM **** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
1044 11E2 REM **** (RETURNED IN TEXT$ STRING VARIABLE) ***
1044 11E2 REM ****
1044 11E2 REM **** INPUT : SHEET RECORD, RECORD LENGTH ***
1044 11E2 REM ****
1044 11E2 REM **** OUTPUT : 'TEXT$' TRING VARIABLE ***
1044 11E2 REM ****
1044 11E2 REM **** RESERVED LINE ***
1044 11E2 REM **** NUMBERS : 8001-8500 ***
1044 11E2 REM ****
1044 11E2
1044 11E2 'DECODE THE HEADER ONLY
1044 11E2 8001 POINTER = 0
1048 11E2 RECORDPTR = VARPTR(SHEETREC(0))
1052 11E4 FOR J8000 = 1 TO 21
1059 11E4 8002 TEXT$= TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
1077 11E4 POINTER=POINTER+1
107F 11E4 NEXT J8000
108E 11E6 PROGRAM$= LEFT$(TEXT$,3)
109D 11E6 BATCH$= MID$(TEXT$,4,3)
10AF 11E6 SERIAL$= MID$(TEXT$,7,4)
10C1 11F2 RUNIDS= MID$(TEXT$,11,1)
1003 11F6 FORM$= MID$(TEXT$,12,2)
10E5 11FA POCKET$= MID$(TEXT$,14,1)

```

KACP940
SHORT FORM

PAGE 17
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
10F7	11FE	SCANERR1\$=MID\$(TEXT\$,16,2)
1109	1202	SCANERR2\$=MID\$(TEXT\$,18,2)
1118	1206	SCANERR3\$=MID\$(TEXT\$,20,2)
112D	120A	GOTO 8500
1131	120A	
1131	120A	8050 REM CHECK FOR END OF JOB/END OF BATCH
1132	120A	IF PROGRAMS = PGMIDS THEN GOTO 8500
1144	120A	LPRINT STRING\$(80,"")
1152	120A	LPRINT
115A	120A	LPRINT "RECORDS PROCESSED ... ";SERIAL\$
1167	120A	LPRINT "STARTED AT ";BTIM\$
1174	120A	LPRINT "ENDED AT ";TIM\$
1181	120A	LPRINT CHR\$(12)
118C	120A	GOTO 30000
1190	120A	
1190	120A	8070 REM CHECK FOR SCANNER ERRORS
1191	120A	IF POCKET\$ = " " GOTO 8500
11A3	120A	LPRINT LITHOS;
11AB	120A	LPRINT " ... SCANNER ERRORS : ";
11B3	120A	LPRINT SCANERR1\$;" / ";
11C0	120A	LPRINT SCANERR2\$;" / ";
11CD	120A	LPRINT SCANERR3\$
11D5	120A	LN=LN+1
11DD	120C	GOTO 999
11E1	120C	
11E1	120C	8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
11E2	120C	LOCATE 5,1:PRINT "PROGRAM ";PROGRAMS;
11FC	120C	
11FC	120C	PRINT " BATCH ";BATCH\$;
1209	120C	PRINT " RUN ";RUNIDS;
1216	120C	PRINT " FORM ";FORM\$;
1223	120C	PRINT " POCKET ";POCKET\$
1230	120C	GOTO 8500
1234	120C	
1234	120C	8200 REM DECODE THE RESPONSE POSITIONS
1235	120C	POINTER = 21
123C	120C	RECORDPTR = VARPTR(SHEETREC(0))
1243	120C	FOR J8000 = 22 TO RECORDLENGTH
1250	120E	8202 TEXT\$ = TEXT\$+CHR\$(PEEK(RECORDPTR + POINTER))
126E	120E	POINTER=POINTER+1
1276	120E	NEXT J8000
1287	120E	
1217	120E	8500 RETURN
128A	120E	
128A	120E	REM ----- END OF RXXS8000.SUB -----
128A	120E	
128A	120E	9000 REM \$INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
128B	120E	REM *****
128B	120E	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
128B	120E	REM **** SKIP COLE ****
128B	120E	REM **** PROGRAM NAME : RACS9000.SUB ****
128B	120E	REM **** SCANNER PROGRAM # : ALL ****
123B	120E	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
128B	120E	REM **** CONTROLS THE SCANNER I/O ****

RACP940
SHORT FORM

PAGE 18
07-06-87
15:07:19

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
1288	120E	REM *****
1288	120E	REM ***** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS *****
1288	120E	REM ***** COMMUNICATIONS MANUAL AND THE *****
1288	120E	REM ***** PRE-RELEASED SOFTWARE GUIDE *****
1288	120E	REM *****
1288	120E	REM *****
1288	120E	REM ***** RESERVED LINE *****
1288	120E	REM ***** NUMBERS : 9001-9100 *****
1288	120E	REM *****
1288	120E	*****
1288	120E	*****
1288	120E	REM *****
1288	120E	REM ***** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER *****
1288	120E	REM ***** ARGUMENTS: PRESET ... SEE BELOW *****
1288	120E	REM *****
1288	120E	9001 REM
128C	120E	PROTOCOL(0) = 9600 'BAUD RATE
1293	120E	PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
129A	120E	PROTOCOL(2) = 8 'DATA BITS
12A1	120E	PROTOCOL(3) = 1 'STOP BITS
12A8	120E	PROTOCOL(4) = 2 'RS-232 PORT
12AF	120E	PROTOCOL(5) = 0 'WRITE TIME-OUT
12B6	120E	PROTOCOL(6) = 0 'READ TIME-OUT
12BD	120E	
12BD	120E	ERRSTAT\$ = SPACES(60)
12C9	120E	ARGPTR = VARPTR(PROTOCOL(0))
12D0	1210	CALL SETUP (ARGPTR,ERRSTAT\$)
12E1	1210	ERRMSG\$=""
12EA	1210	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="SETUP ERROR "+ERRSTAT\$
1306	1210	GOTO 9100
130A	1210	
130A	1210	REM *****
130A	1210	REM ***** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER *****
130A	1210	REM ***** ARGUMENTS: CNTRLOPT *****
130A	1210	REM ***** CNTRLOPT = 1 = START SCANNER (SI) *****
130A	1210	REM ***** CNTRLOPT = 2 = STOP SCANNER (SO) *****
130A	1210	REM ***** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) *****
130A	1210	REM ***** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) *****
130A	1210	REM ***** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" *****
130A	1210	REM ***** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" *****
130A	1210	REM ***** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) *****
130A	1210	REM ***** CNTRLOPT = 8 = REQUEST STATUS (ESC) *****
130A	1210	REM *****
130A	1210	9010 REM
130B	1210	ERRSTAT\$ = SPACES(60)
1317	1210	CALL CNTRL (CNTRLOPT,ERRSTAT\$)
1328	1210	ERRMSG\$=""
1331	1210	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="CONTROL ERROR "+ERRSTAT\$
134D	1210	GOTO 9100
1351	1210	
1351	1210	REM *****
1351	1210	REM ***** SUBROUTINE 9020 - SCAN SHEET CALL *****
1351	1210	REM *****
1351	1210	REM ***** ARGUMENTS: READTYPE *****

Offset	Data	Source Line
1351	1210	REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
1351	1210	REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
1351	1210	REM ****
1351	1210	REM **** ARGUMENTS: RECORDLENGTH ****
1351	1210	REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
1351	1210	REM **** TRANSMITTED ****
1351	1210	REM *****
1351	1210	9020 REM
1352	1210	ERRSTAT\$ = SPACES(60)
135E	1210	RECORDPTR = VARPTR(SHEETREC(0))
1365	1210	CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT\$)
137E	1210	ERRMSG\$=""
1387	1210	IF MID\$(ERRSTAT\$,14,3) = "415" THEN ERRMSG\$="ESC"
13A8	1210	GOTO 9100
13AC	1210	
13AC	1210	REM *****
13AC	1210	REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
13AC	1210	REM ****
13AC	1210	REM **** ARGUMENTS: PRINTPOS ****
13AC	1210	REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
13AC	1210	REM **** VALUES = 0 THRU 90 ****
13AC	1210	REM ****
13AC	1210	REM **** ARGUMENTS: PSTRING\$ ****
13AC	1210	REM **** TEXT TO BE PRINTED ON THE FORM ****
13AC	1210	REM ****
13AC	1210	REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
13AC	1210	REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
13AC	1210	REM *****
13AC	1210	9030 REM
13AD	1210	ERRSTAT\$ = SPACES(60)
13B9	1210	RECORDPTR = VARPTR(SHEETREC(0))
13C0	1210	CALL TPRINT(PRINTPOS,PSTRING\$,ERRSTAT\$)
13D5	1216	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="PRINT ERROR "+ERRSTAT\$
13F1	1216	GOTO 9100
13F5	1216	
13F5	1216	9100 RETURN
13F8	1216	REM -----END OF SUBROUTINE RACS9000.SUB -----
13F8	1216	
13F8	1216	REM END OF SUBROUTINES *****
13F8	1216	
13F8	1216	25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
13F9	1216	LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1401	1216	LPRINT "ERASING FILE ";DATFIL\$
140E	1216	BEEP
1412	1216	CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
141E	1216	CLOSE
1422	1216	OPEN DATFIL\$ FOR OUTPUT AS #1
1434	1216	PRINT #1,STRING\$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
1446	1216	CLOSE
144A	1216	
144A	1216	30000 REM
144B	1216	CLOSE
144F	1216	CHAIN "RACP10"
1456	1216	END

RACP940
SHORT FORM

PAGE 20

07-06-87

15:07:19

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

145A 1216
145D 1216

22151 Bytes Available

16962 Bytes Free

0 Warning Error(s)

0 Severe Error(s)

```

*-----*
*
* PROCEDURE LOADDIAG
*
* LOADS THE DIAGNOSIS DATA BASE
* DATA IS IN FILE 'DIAGNOS.DAT'
*
*-----*
FILEDEF DIAGNOS  DISK DIAG.RVD
-CRTCLEAR
MODIFY FILE DIAGNOS
FIXFORM DGNCD/C5 X2 DGNDSC/C50 X13
MATCH DIAG_CODE
    ON NOMATCH INCLUDE
    ON  MATCH CONTINUE
DATA ON DIAGNOS
END

```

```

*-----*
*
* PROCEDURE LOADOTH
*
* LOADS THE OTHER CODE DATA BASE
* DATA IS IN FILE 'OTHER.DAT'
*
*-----*

```

```

FILEDEF OTHER DISK C:OTHER.DAT
-CRTCLEAR
MODIFY FILE OTHER
FIXFORM OTHC/C5 X2 OTHDSC/C67
MATCH OTHER_CODE
  ON NOMATCH INCLUDE
  ON MATCH CONTINUE
DATA ON OTHER
END

```

```

*-----*
*
* PROCEDURE LOADPROC
*
* LOADS THE PROCEDURE DATA BASE
* DATA IS IN FILE 'PROCEDUR.DAT
*
*-----*
FILEDEF PROCEDUR DISK PROC.RVD
-CRTCLEAR
MODIFY FILE PROCEDUR
FIXFORM PRCCD/C5 X2 PRCDSC/C67
MATCH PROC_CODE
    ON NOMATCH INCLUDE
    ON MATCH CONTINUE
DATA ON PROCEDUR
END

```

PAGE 1

07-06-87

15:11:33

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```
001A 0002 REM $LINESIZE: 132
001A 0002 REM $PAGESIZE: 66
001A 0002 REM $TITLE: 'RACP950 '
001A 0002 REM $SUBTITLE: 'SOCIAL WORK SHORT FORM'
001A 0002 REM $PAGE
```

```

Offset  Data      Source Line
001A 0002 REM +-----+
001A 0002 REM | NAME: RACP820          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002 REM | DATE: 14 MAR 87          SOCIAL WORK (BFEA) SHORT FORM |
001A 0002 REM | D R BOLLING          SHORT FORM |
001A 0002 REM |
001A 0002 REM | INCLUDES PREFIX TO LITHO FOR EACH PATIENT ON FORM |
001A 0002 REM +-----+
001A 0002 REM          SOCIAL WORK SHORT FORM INPUT PROGRAM
001A 0002 REM
001A 0002 REM This program reads the SHORT form OMR data, converts various
001A 0002 REM fields, prints an error report and produces the file:
001A 0002 REM
001A 0002 REM          VISIT.DAT
001A 0002 REM
001A 0002 REM for input to FOCUS. NOTE THAT THIS FILE IS OPENED FOR APPEND
001A 0002 REM each time the program is run. Thus, if the file does not exist,
001A 0002 REM records will be added to the front. If the file exists, records
001A 0002 REM will be added to the end of the current file. It is intended that
001A 0002 REM the FOCUS DIALOGUE MANAGER ROUTINE which loads the data will delete
001A 0002 REM the data file after the load has been successfully accomplished.
001A 0002 REM
001A 0002 REM
001A 0002 REM If there is no valid user logged at the time of execution, this
001A 0002 REM program will chain to the logon program RACP05, otherwise,
001A 0002 REM the program chains to program RACP10 on exit.
001A 0002
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD          DIMENSION DEFINITIONS *
001A 0002 '* Date: 28 Feb 84          Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM DIMENSION STATEMENTS UNIQUE TO THIS PROGRAM.....
001A 0002
001A 0002 DIM SHEETREC(1750) '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002 DIM PROTOCOL(7) '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002 DIM ED.MSG$(30) '(ERROR MESSAGES FROM EDIT ROUTINES)
001A 0002 DIM CLINIC2.PFX$(5) '(PREFIX -A B D F S- FOR CLINIC #2)
001A 0002 DIM PROCED$(05) '(PROCEDURE TABLE FOR SHORT FORM)
001A 0002 DIM DIAG.TAB$(6) '(DIAGNOSIS TABLE - PROBLEM)

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

001A 0002      DIM PROVIDER.TIMES(8) '(PROVIDER TIME TABLE)
001A 0002      DIM SPECODS(9)      '(SPECIFIC PREASSIGNED CLINIC CODES)
001A 0002      DIM SSN.CODS(4)      '
001A 0002      DIM FMP.CODS(4)      '
001A 0002      DIM REF.CODS(4)      '
001A 0002      DIM PBM.CODS(4)      '
001A 0002      DIM TOT.PROB(4)      '
001A 0002      DIM PR1.TIMS(4)      ' . ARRAYS TO HOLD MULTIPLE VISITS
001A 0002      DIM TOT.PROC(4)      '
001A 0002      DIM SPE.BUFS(4)      '
001A 0002      DIM TOT.SPE(4)      '
001A 0002      DIM GROUP1$(4,06)    '(PATIENT/PROCEDURE GROUP PROV 1)
001A 0002      DIM HOLDS(06)        '(HOLD AREA FOR SUBROUTINE 6000)
001A 0002
001A 0002      REM $INCLUDE: 'RACCHN.MOD'      REM INCLUDE THE COMMON AREA DEFINITION
001A 0002      *****
001A 0002      '* NAME: RACCHN.MOD              COMMON AREA DEFINITION      *
001A 0002      '* Date: 28 Feb 84              Written by: Floyd Cole      *
001A 0002      *****
001A 0002      ' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A 0002      ' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A 0002      '
001A 0002      ' This program segment may be modified, but all files containing
001A 0002      ' an include for this segment must be re*compiled in order to
001A 0002      ' affect the changes made here.
001A 0002      '
001A 0002      ' *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002      COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELLS 'BASIC SCREEN COLORS
001A 0002      COMMON HEADERS      '21 CHARACTER SCANNER HEADER INFO
001A 0002      COMMON TEXT$      ' ' AINING CHARACTERS FROM SCANNER
001A 0002      COMMON PGMIDS      'PROGRAM OR FORM ID
001A 0002      COMMON MOLENGTH( )      'DAYS IN THE MONTH
001A 0002      COMMON USERS( )
001A 0002      ' *****END OF COMMON DEFINITION*****
001A 0002
001A 0002      REM $INCLUDE: 'RACDEF.MOD'      REM INCLUDE THE DEFAULT DEFINITIONS
001A 0002      *****
001A 0002      '* NAME: RACP01.DEF              DEFAULT DEFINITIONS      *
001A 0002      '* Date: 28 Feb 84              Written by: Floyd Cole      *
001A 0002      *****
001A 0002      ' Variables used in common that have a default value on start*up
001A 0002      ' will be held in this file. It is an included file so it cannot
001A 0002      ' be run in a stand*alone mode. In normal operation, this file
001A 0002      ' should be 'included' in the main program only (RACP10.BAS).
001A 0002      '
001A 0002      ' This program segment may be modified, but all files containing
001A 0002      ' an include for this segment must be re*compiled in order to
001A 0002      ' affect the changes made here.
001A 0002      '
001A 0002      ' *****START OF DEFAULT DEFINITION*****
001A 0002      FORE = 15      'FOREGROUND COLOR = INTENSE WHITE

```

Offset	Data	Source Line
0050	1022	BACK = 1 'Background Color = Light Blue
0057	1022	BORD = 4 'BORDER = RED
005E	1024	HIDE = 4 'ALTERNATE COLOR = RED
0065	1024	EFORE= 14 'ERROR FOREGROUND DISPLAY
006C	1024	EBACK= 0 'ERROR BACKGROUND DISPLAY
0073	1024	BELL\$ = CHR\$(7) 'Sound the bell
007F	1024	
007F	1024	MOLENGTH(1) = 31 'JAN
0086	1024	MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
008D	1024	MOLENGTH(3) = 31 'MAR
0094	1024	MOLENGTH(4) = 30 'APR
009B	1024	MOLENGTH(5) = 31 'MAY
00A2	1024	MOLENGTH(6) = 30 'JUN
00A9	1024	MOLENGTH(7) = 31 'JUL
00B0	1024	MOLENGTH(8) = 31 'AUG
00B7	1024	MOLENGTH(9) = 30 'SEP
00BE	1024	MOLENGTH(10) = 31 'OCT
00C5	1024	MOLENGTH(11) = 30 'NOV
00CC	1024	MOLENGTH(12) = 31 'DEC
00D3	1024	
00D3	1024	DATEERR\$(0) = " "
00DC	1024	DATEERR\$(1) = "INVALID MONTH"
00E5	1024	DATEERR\$(2) = "INVALID DAY "
00EE	1024	DATEERR\$(3) = "DAY TOO LARGE FOR MONTH CODED"
00F7	1024	
00F7	1024	MAXLENGTH = 80 'MAXIMUM LENGTH OF OUTPUT RECORD
00FE	1026	PAD\$ = " " 'PAD CHARACTER FOR SHORT RECORDS
0107	102A	
0107	102A	' *****END OF DEFAULT DEFINITION*****
0107	102A	
0107	102A	
0107	102A	KEY OFF
0100	102A	
0100	102A	REM *****
0100	102A	REM THE FOLLOWING VARIABLES ARE UNIQUE TO EACH PROGRAM AND MUST
0100	102A	REM BE CHANGED.
0100	102A	REM *****
0100	102A	PGMTITL\$ = "SOCIAL WORK (BFEA) SHORT FORM"
0116	102E	
0116	102E	PGMID\$ = "950" 'VALUE RECEIVED FROM THE SCANNER
011F	102E	'IN HEADER VARIABLE 'PROGRAMS'
011F	102E	
011F	102E	DATFIL\$ = "VISIT.DAT" 'FILE TO BE INPUT TO FOCUS
0128	1032	
0128	1032	REM LENGTH OF STRING RECEIVED FROM THE OMR....
0128	1032	HEADER = 21
012F	1034	RESPONSE= 162
0136	1036	RECORDLENGTH = HEADER + RESPONSE
0141	1038	
0141	1038	N.PROC = 05 ' NUMBER OF PROCEDURES FOR THIS FORM
0148	103A	
0148	103A	REM *****
0148	103A	
0148	103A	BTIME\$=TIME\$ 'SCAN START TIME

Offset Data Source Line

```
0151 103E
0151 103E REM *** ENCOUNTER FORM CLINIC PREFIX TABLE ***
0151 103E REM CLINIC #2
0151 103E CLINIC2.PFX$(0)=" "
015A 103E CLINIC2.PFX$(1)="A"
0163 103E CLINIC2.PFX$(2)="B"
016C 103E CLINIC2.PFX$(3)="D"
0175 103E CLINIC2.PFX$(4)="F"
017E 103E CLINIC2.PFX$(5)="S"
0187 103E
0187 103E REM *** ENCOUNTER FORM PROCEDURE TABLE ***
0187 103E PROCED$(0)=" " : PROCED$(02) = "02404" : PROCED$(04)= "02413"
01A2 103E PROCED$(1)="03010" : PROCED$(03) = "02412" : PROCED$(05)= "90601"
018D 103E
018D 103E REM *** PROBLEM -DX TABLE ***
018D 103E DIAG.TAB$(0)=" " : DIAG.TAB$(3)="$S2104"
01CF 103E DIAG.TAB$(1)="$S2159" : DIAG.TAB$(4)="$S2266"
01E1 103E DIAG.TAB$(2)="$S2296" : DIAG.TAB$(5)="$S2297"
01F3 103E DIAG.TAB$(6)="$S2109"
01FC 103E
01FC 103E REM *** PROVIDER TIME TABLE ***
01FC 103E PROVIDER.TIMES$(00)="$000" ' NO TIME
0205 103E PROVIDER.TIMES$(01)="$005" ' 5 MINUTES
020E 103E PROVIDER.TIMES$(02)="$010" '10 MINUTES
0217 103E PROVIDER.TIMES$(03)="$015" '15 MINUTES
0220 103E PROVIDER.TIMES$(04)="$020" '20 MINUTES
0229 103E PROVIDER.TIMES$(05)="$030" '30 MINUTES
0232 103E PROVIDER.TIMES$(06)="$040" '40 MINUTES
0238 103E PROVIDER.TIMES$(07)="$050" '50 MINUTES
0244 103E PROVIDER.TIMES$(08)="$060" ' 1 HOURS
024D 103E
024D 103E
024D 103E REM INCLUDE: 'UCABAMC.OPT' INCLUDE OUTPATIENT UCA TABLE
024D 103E
024D 103E REM $PAGE
```

Offset	Data	Source Line
0240	103E	GOSUB 1000 'MAKE SURE THEY ARE LOGGED ON
0252	103E	CLS
0256	103E	GOSUB 7000 'PRINT SCREEN HEADING
0258	103E	
0258	103E	REM *****
0258	103E	REM **** OPEN FILE TO CONTAIN SCANNED DATA ****
0258	103E	REM *****
0258	103E	REM
0258	103E	OPEN DATFIL\$ FOR APPEND AS #1
0260	103E	
0260	103E	REM *****
0260	103E	REM **** CLEAR AND DISPLAY PROGRAM SCREEN ****
0260	103E	REM *****
0260	103E	LPRINT CHR\$(15);
0278	103E	WIDTH "LPT1:",160
0282	103E	PAGE = 0 : GOSUB 7100 'LINE PRINTER HEADING
028E	1040	COLOR 14
0295	1040	LOCATE 11,26 : PRINT "SOCIAL WORK SHORT FORM"
02AA	1040	COLOR FORE,BACK,BORD
02C0	1040	
02C0	1040	REM *****
02C0	1040	REM **** COMMUNICATIONS SETUP ****
02C0	1040	REM *****
02C0	1040	REM PROTOCOL
02C0	1040	GOSUB 9001
02C5	1040	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
02DF	1044	
02DF	1044	REM START SCANNER (SI)
02DF	1044	CNTRLOPT =1 :GOSUB 9010
02EB	1046	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0305	1046	
0305	1046	LOCATE 22,25:PRINT "PRESS 'ESC' TO TERMINATE SCANNING "
031A	1046	READYTYPE=3 'FIRST TIME IN.. SCANNER IS STARTED..
0321	1048	
0321	1048	REM *****
0321	1048	REM **** SET SCAN SHEET CALL ****
0321	1048	REM *****
0321	1048	REM
0321	1048	10 REM - RETURN POINT TO READ NEXT SHEET
0322	1048	
0322	1048	AS=INKEY\$
032B	104C	IF AS=CHR\$(27) THEN GOTO 25000
0341	104C	
0341	104C	GOSUB 9020 'SCAN SUBROUTINE - GET A RECORD
0346	104C	IF MID\$(ERRSTAT\$,14,3)="415" THEN GOTO 25000
0362	1050	
0362	1050	TEXT\$="" 'CLEAR THE INPUT AREA
0368	1050	GOSUB 8000 'DECODE HEADER
0370	1050	GOSUB 8050 'CHECK FOR END OF JOB/END OF BATCH
0375	1050	GOSUB 8200 'DECODE THE RESPONSE POSITIONS
037A	1050	LITHOS = MID\$(TEXT\$,22,8)
038C	1054	GOSUB 8070 'CHECK FOR SCANNER ERRORS
0391	1054	GOSUB 8100 'PRINT THE DATA ON THE SCREEN

Offset	Data	Source Line
0396	1054	
0396	1054	REM \$INCLUDE: 'RACH950.MO1' INCLUDE THE SHORT FORM REFORMAT/EDIT MOD
0396	1054	REM *****
0396	1054	REM **** AMBULATORY CARE INFORMATION SYSTEM 14 MAR 87 ****
0396	1054	REM **** D R BOLLING ****
0396	1054	REM **** MODULE NAME : RACH950.MO1 ****
0396	1054	REM **** SCANNER PROGRAM # : 950- SOCIAL WORK (BFEA) SHORT ****
0396	1054	REM ****
0396	1054	REM **** PURPOSE : REFORMAT/EDIT THE ENCOUNTER ****
0396	1054	REM **** SHORT FORM OMR RECORD. ****
0396	1054	REM *****
0396	1054	REM **** RESERVED LINE NUMBERS 100-199 ****
0396	1054	REM *****
0396	1054	
0396	1054	N.ERR =0 'COUNTS THE NUMBER OF ERRORS
0390	1056	
0390	1056	REM *** LITHO CODE DONE IN BAS PROGRAM ***
0390	1056	
0390	1056	REM *** CLINIC ID ***
0390	1056	100 CL1.CODS="BFEA" 'DEFAULT CLINIC CODE
03A6	105A	
03A6	105A	REM *** VISIT DATE ***
03A6	105A	102 CMS=MID\$(DATE\$,1,2) 'CURRENT MONTH
03B8	105E	YRS=MID\$(DATE\$,9,2) 'CURRENT YEAR
03CA	1062	XS=MID\$(TEXT\$,31,4) 'MONTH AND DAY FROM FORM
030C	1066	IF LEFT\$(XS,2)<=CMS THEN 104 'OK, USE THIS YEAR
03F1	1066	YRS=RIGHT\$(STR\$(VAL(YRS)-1),2) 'USE LAST YEAR
0411	1066	104 VDATES = YRS + XS
041F	106A	D1\$=MID\$(XS,3,1)
0431	106E	D2\$=MID\$(XS,4,1)
0443	1072	IF D1\$=" " AND D2\$<>" " THEN RT.5000=2 : GOTO 105
0471	1074	IF D1\$<>" " AND D2\$=" " THEN RT.5000=2 : GOTO 105
049F	1074	'EDIT VISIT DATE
049F	1074	CK.5000\$=VDATES
04A8	1078	GOSUB 5000 'DATE CHECK
04AD	1078	CK.5010\$=VDATES
04B6	107C	GOSUB 5010 'NUMERIC STRING CHECK
04B8	107C	IF RT.5000 = 0 AND RT.5010 = 0 THEN GOTO 106
04DF	107E	105 N.ERR=N.ERR+1
04E7	107E	ED.MSG\$(N.ERR)="TODAYS DATE " + DATEERR\$(RT.5000)
0500	107E	
0500	107E	REM *** PROVIDER 1 ID (PREFIX + NUM) ***
0500	107E	106 PROV1.PFX\$ = MID\$(TEXT\$,35,1)
051F	1082	PROV1.NUM\$ = MID\$(TEXT\$,36,4)
0531	1086	PR1.CODS=PROV1.PFX\$ + PROV1.NUM\$
053F	108A	
053F	108A	SSN.OFFSET = 40
0546	108C	FMP.OFFSET = 49
054D	108E	RPF.OFFSET = 51
0554	1090	REF.OFFSET = 52
055B	1092	TIM.OFFSET = 55
0562	1094	UPR.OFFSET = 56
0569	1096	PBM.OFFSET = 61
0570	1098	PRC.OFFSET = 62

Offset	Data	Source Line
0577	109A	SPE.OFFSET = 67
057E	109C	TOT.OFFSET = 36
0585	109E	POINTER = 0
058C	10A0	
058C	10A0	
058C	10A0	REM *** REPEAT THE FOLLOWING CODE 4 TIMES ***
058C	10A0	FOR I910= 1 TO 4
0593	10A0	
0593	10A0	REM *** SSN AND FMP ***
0593	10A0	110 SSN.COD\$(I910)=MID\$(TEXT\$,SSN.OFFSET+POINTER,9)
0586	10A2	FMP.COD\$(I910)=MID\$(TEXT\$,FMP.OFFSET+POINTER,2)
05D9	10A2	REM BYPASS IF NO SSN OR FMP FOR THIS PART
05D9	10A2	IF SSN.COD\$(I910)+FMP.COD\$(I910)=STRING\$(11," ") THEN 136
0606	10A2	
0606	10A2	REM *** REFERRAL CODE ***
0606	10A2	112 X=VAL(MID\$(TEXT\$,RPF.OFFSET+POINTER,1))
0621	10A4	REF.COD\$(I910)=CLINIC2.PFX\$(X)+MID\$(TEXT\$,REF.OFFSET+POINTER,3)
0655	10A4	
0655	10A4	REM *** TIME SPENT ***
0655	10A4	114 X=VAL(MID\$(TEXT\$,TIM.OFFSET+POINTER,1))
0670	10A4	PR1.TIMS(I910)=PROVIDER.TIMES(X)
068D	10A4	IF PR1.TIMS(I910)<>"000" THEN 116 'GO IF THERE IS A TIME
06A5	10A4	N.ERR=N.ERR+1
06AD	10A4	ED.MSG\$(N.ERR)="NO TIME CODED FOR PART "+STR\$(I910)
06CA	10A4	
06CA	10A4	REM *** ADDITIONAL PROCEDURE
06CA	10A4	116 ADDPS=MID\$(TEXT\$,UPR.OFFSET+POINTER,5)
06E1	10A8	
06E1	10A8	REM *** PROBLEM ***
06E1	10A8	118 X = VAL(MID\$(TEXT\$,PBM.OFFSET+POINTER,1))
06FC	10A8	PBM.COD\$(I910)=DIAG.TAB\$(X)
0719	10A8	
0719	10A8	REM *** PROCEDURE CODES FOR PROV 1 ***
0719	10A8	120 X\$ = MID\$(TEXT\$,PRC.OFFSET+POINTER,N.PROC)
0731	10A8	GOSUB 6000 'DECODE THE X\$ STRING
0736	10A8	IF TOT = 0 THEN GOTO 122
0745	10AA	FOR I910.A= 1 TO TOT
0752	10AC	PTR=VAL(HOLD\$(I910.A))
0768	1080	GROUP1\$(I910,I910.A)=PROCD\$(PTR)
078A	1080	NEXT I910.A
0798	1080	
0798	1080	REM ... ADD UNLISTED CODE IF THERE TO PROV 1 ...
0798	1080	122 IF ADDPS="" THEN 124
07A9	1080	TOT=TOT+1
07B1	1080	GROUP1\$(I910,TOT)=ADDPS
07CA	1080	
07CA	1080	124 TOT.PROC(I910)=TOT
07D9	1080	
07D9	1080	REM *** SPECIFIC PREASSIGNED CL CODES ***
07D9	1080	126 INP.STOS=MID\$(TEXT\$,SPE.OFFSET+POINTER,9)
07F0	1084	GOSUB 5700 'CONVERT ARRAY
07F5	1084	SPE.BUF\$(I910)=BUF.STOS 'UP TO 9 TWO DIGIT CODES
0809	1088	
0809	1088	130 POINTER = POINTER + TOT.OFFSET

RACP950

SOCIAL WORK SHORT FORM

PAGE 9

07-06-87

15:11:33

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
0814	1088	136 NEXT 1910
0826	1088	199 REM
0827	1088	
0827	1088	REM -----END OF MODULE RXM950.M01-----
0827	1088	
0827	1088	
0827	1088	IF N.ERR = 0 THEN GOTO 997
0836	1088	LPRINT "LITHO # ";LITHOS;" ... ERRORS"
0848	1088	FOR 1997 = 1 TO N.ERR
0855	108A	LPRINT USING "### ";1997;
0861	108C	LPRINT "=> ";ED.MSG\$(1997)
0877	108C	NEXT 1997
0888	108C	LN.COUNT = LN.COUNT + N.ERR + 1
0894	108E	CNTRLOPT = 6
0898	108E	GOSUB 9010 'REJECT THE FORM
08A0	108E	GOTO 998 'BYPASS THE DISK WRITER....
08A4	108E	
08A4	108E	997 REM \$INCLUDE: 'RACH950.M02' REM INCLUDE THE BASE ENCOUNTER FORM DISK WRITER
08A5	108E	REM *****
08A5	108E	REM **** AMBULATORY CARE INFORMATION SYSTEM 14 MAR 87 ****
08A5	108E	REM **** D R BOLLING ****
08A5	108E	REM **** MODULE NAME : RACH950.M02 ****
08A5	108E	REM **** SCANNER PROGRAM # : 950- SOCIAL WORK SHORT FORM ****
08A5	108E	REM ****
08A5	108E	REM **** PURPOSE : CREATE AND WRITE THE DISK ****
08A5	108E	REM **** RECORD FOR INPUT TO FOCUS ****
08A5	108E	REM ****
08A5	108E	REM **** PROGRAM ADDS PREFIX TO LITHO FOR EACH PATIENT ****
08A5	108E	REM *****
08A5	108E	REM **** RESERVED LINE NUMBERS 200-299 ****
08A5	108E	REM *****
08A5	108E	
08A5	108E	REM BUILD THE OUTPUT RECORD
08A5	108E	
08A5	108E	GOSUB 270 'BUILD THE RECORD KEY
08AA	108E	
08AA	108E	REM *****
08AA	108E	REM **** RECORD TYPE "1" - RECKEY PLUS DATA AND DX1 CODE ****
08AA	108E	REM *****
08AA	108E	
08AA	108E	FOR 1950 = 1 TO 4
08B1	108E	RECOUT\$ = "9501" + RECKEY\$ 'INITIALIZE
08BF	10C6	PTID\$=SSN.COD\$(1950)+FMP.COD\$(1950)
08DE	10CC	IF PTID\$=STRING\$(11," ") THEN 248 'NO MORE TO DO IF BLANK
08F3	10CC	RECOUT\$=RECOUT\$+RIGHT\$(STR\$(1950),1)+RIGHT\$(LITHOS,7)
0918	10CC	MID\$(RECOUT\$,20,11)=PTID\$
0927	10CC	RECOUT\$ = RECOUT\$ + "95" 'ADD FORM NUMBER
0933	10CC	
0933	10CC	REC0D1\$="" 'INITIALIZE
093C	1000	
093C	1000	REM *** VISIT COUNT ***
093C	1000	REC0D1\$=REC0D1\$+"1"
0948	1000	
0948	1000	REM *** TIME PROV1 ***

```

Offset Data Source Line
0948 1000 RECOD1$=RECOD1$+PR1.TIMS(1950)
0950 1000
0950 1000 REM *** SEC PROV - SEC PROV TIME - SEC PROV REAS -APP STAT ***
0950 1000 RECOD1$=RECOD1$+STRINGS(10," ")
096F 1000
096F 1000 REM *** REFERRAL CODE ***
096F 1000 RECOD1$=RECOD1$+REF.CODS(1950)
0984 1000
0984 1000 REM *** REMAINING AREA FILL ***
0984 1000 RECOD1$=RECOD1$+STRINGS(11," ")
0996 1000
0996 1000 REM *** PRIM DX ***
0996 1000 RECOD1$=RECOD1$+PRM.CODS(1950)
09AB 1000 RECOUNT$=RECOUNT$+RECOD1$
09B7 1000 GOSUB 280
09BC 1000 PRINT #1,RECOUNT$
09C7 1000
09C7 1000 NEXT 1950
09D9 1000
09D9 1000 248 REM END OF TYPE 1 RECORD PROCESSING
09DA 1000
09DA 1000 REM *****
09DA 1000 REM **** RECORD TYPE "2" - RECKEY PLUS PROCEDURE CODE ****
09DA 1000 REM *****
09DA 1000 REM RECOUNT$ ="9502"+RECKEY$ 'TRANSACTION IDENTIFIER
09DA 1000
09DA 1000 FOR 1950 = 1 TO 4
09E1 1000 PTID$=SSN.COD$(1950)+FMP.COD$(1950)
0A00 1000 IF PTID$=STRINGS(11," ") THEN 251 'NO MORE TO DO IF BLANK
0A15 1000 FOR 1910.A = 1 TO TOT.PROC(1950)
0A29 1002 RECOUNT$ = "9502" + RECKEY$ + RIGHTS$(STR$(1950),1) + RIGHTS$(LITHOS,7)
0A59 1002 MID$(RECOUNT$,20,11)=PTID$
0A68 1002 RECOUNT$ = RECOUNT$ + "95" 'ADD FORM NUMBER
0A74 1002 RECOUNT$ =RECOUNT$ + "1" + GROUP1$(1950,1910.A)
0A98 1002 GOSUB 280
0A9D 1002 PRINT #1,RECOUNT$
0AA8 1002 NEXT 1910.A
0ABC 1002 250 NEXT 1950
0ACE 1002
0ACE 1002 251 REM END OF TYPE 2 RECORD PROCESSING
0ACF 1002
0ACF 1002 REM *****
0ACF 1002 REM **** RECORD TYPE "3" - RECKEY PLUS SPEC PRE CLINIC CODES ****
0AF 1002 REM *****
0ACF 1002 FOR 1950 = 1 TO 4
0AD6 1002 PTID$=SSN.COD$(1950)+FMP.COD$(1950)
0AF5 1002 IF PTID$=STRINGS(11," ") THEN 256 'NO MORE TO DO IF BLANK
0B0A 1002 IF LEN$(SPE.BUF$(1950))=0 THEN 254
0B20 1002 RPOINT=1
0B27 1004 252 RECOD3$=RIGHT$(MID$(SPE.BUF$(1950),RPOINT,2),1)
0B49 1008 IF RECOD3$="" THEN 253
0B57 1008 RECOUNT$ = "9503" + RECKEY$ + RIGHTS$(STR$(1950),1) + RIGHTS$(LITHOS,7)
0B87 1008 MID$(RECOUNT$,20,11)=PTID$
0B96 1008 RECOUNT$ = RECOUNT$ + "95" 'ADD FORM NUMBER

```

Offset	Data	Source Line
08A2	1008	RECOUT\$ = RECOUT\$ + REC003\$
08AE	1008	GOSUB 280
08B3	1008	PRINT #1, RECOUT\$
08BE	1008	
08BE	1008	253 RPOINT = RPOINT + 2
08C7	1008	IF RPOINT < LEN(SPE.BUF\$(1950)) THEN 252
08DF	1008	
08DF	1008	254 NEXT 1950
08F1	1008	
08F1	1008	256 REM END OF TYPE 3 RECORDS
08F2	1008	
08F2	1008	GOTO 299
08F6	1008	
08F6	1008	REM *****
08F6	1008	REM **** SUBROUTINE 270 - BUILD THE RECORD KEY ****
08F6	1008	REM *****
08F6	1008	270 RECKEY\$=""
08FF	1008	
08FF	1008	REM *** CLINIC ID (PREFIX + CODE) ***
08FF	1008	RECKEY\$ = CL1.COD\$
0C08	1008	
0C08	1008	REM *** VISIT DATE ***
0C08	1008	RECKEY\$ = RECKEY\$ + VDATE\$
0C14	1008	
0C14	1008	REM *** PROVIDER ID (PREFIX + NUM) ***
0C14	1008	RECKEY\$ = RECKEY\$ + PR1.COD\$
0C20	1008	
0C20	1008	REM *** PATIENT ID (SSN+PMP) ***
0C20	1008	RECKEY\$ = RECKEY\$ + STRING\$(11, " ")
0C32	1008	
0C32	1008	REM *** LITHO CODE ***
0C32	1008	REM DO LITHO IN LOOP ABOVE
0C32	1008	REM RECKEY\$ = RECKEY\$ + LITHO\$
0C32	1008	
0C32	1008	RETURN
0C35	1008	
0C35	1008	REM *****
0C35	1008	REM **** SUBROUTINE 280 - PAD THE RECORD TO MAXLENGTH ****
0C35	1008	REM *****
0C35	1008	280 PAD=MAXLENGTH - LEN(RECOUT\$) 'FIND OUT HOW SHORT THE RECORD IS
0C43	100A	RECOUT\$ = RECOUT\$ + STRING\$(PAD, PAD\$) 'PAD THE RECORD WITH FILL CHAR
0C56	100A	RETURN
0C59	100A	
0C59	100A	
0C59	100A	299 REM
0C5A	100A	
0C5A	100A	REM -----END OF MODULE RXXM950.M02-----
0C5A	100A	
0C5A	100A	998 REM CONTINUE
0C5B	100A	
0C5B	100A	999 READTYPE = 2
0C62	100A	IF LN.COUNT > 48 THEN GOSUB 7100 'PRINTER HEADING
0C72	100A	GOTO 10
0C76	100A	

```

Offset  Data  Source Line
-----
0C76 100A REM END OF SCAN/DECODE/WRITE LOOP =====
0C76 100A
0C76 100A 1000 REM $INCLUDE: 'RACS1000.SUB' INCLUDE THE VERIFY LOGON SUB
0C77 100A REM *****
0C77 100A REM * NAME: RACS1000 LOGON VERIFICATION SUBROUTINE *
0C77 100A REM * Date: 28 Feb 84 PATIENT REGISTRATION PROGRAM *
0C77 100A REM *****
0C77 100A REM PATIENT OMR INPUT PROGRAM *
0C77 100A REM *
0C77 100A REM This program verifies user is logged on properly. If there is no *
0C77 100A REM valid user logged on at the time of execution, this subroutine will*
0C77 100A REM chain to the logon program RACP05, otherwise a return is issued. *
0C77 100A REM *****
0C77 100A REM RESERVED LINE NUMBERS ARE 1001 THRU 1010
0C77 100A REM *****
0C77 100A 1001 OPEN "I",1,"RACLOG.DAT"
0C89 100A IF EOF(1) THEN 1002 'MAKE THEM LOG ON FIRST
0C97 100A INPUT #1,USERS(1),DT$,TM$,PID$
0C88 10E6 IF USERS(1) = "" THEN 1002 'MAKE THEM LOG ON FIRST
0CC6 10E6 IF USERS(1) = "*****" THEN 1002 'MAKE THEM LOG ON FIRST
0CD4 10E6 CLOSE 1
0CDB 10E6 SCREEN 0,1,0,0
0CF1 10E6 COLOR FORE,BACK,BORD
0D07 10E6 CLS
0D08 10E6 RETURN
0D0E 10E6
0D0E 10E6 1002 CLOSE
0D12 10E6 CHAIN "RACP05"
0D19 10E6 '=====END OF LOGON VERIFY SUBROUTINE 1000=====
0D19 10E6
0D19 10E6 2000 REM $INCLUDE: 'RACS2000.SUB' INCLUDE THE REPLY/DELAY SUB
0D1A 10E6 REM *****
0D1A 10E6 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0D1A 10E6 REM **** SKIP COLE ****
0D1A 10E6 REM **** SUBROUTINE NAME : RACS2000.SUB ****
0D1A 10E6 REM **** SCANNER PROGRAM # : ALL ****
0D1A 10E6 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0D1A 10E6 REM **** SERVERS AS A WAIT AND REPLY ****
0D1A 10E6 REM **** ENTRY MODULE ****
0D1A 10E6 REM **** INPUT : SINGLE KEYBOARD ENTRY ****
0D1A 10E6 REM ****
0D1A 10E6 REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
0D1A 10E6 REM ****
0D1A 10E6 REM **** RESERVED LINE ****
0D1A 10E6 REM **** NUMBERS : 2001-2010 ****
0D1A 10E6 REM *****
0D1A 10E6 2001 REM REPLY FUNCTION
0D1B 10E6 2002 REPLY$=INKEY$ : IF REPLY$="" THEN 2002
0D2F 10EA REPLY=ASC(REPLY$)
0D39 10EC IF REPLY > 90 THEN REPLY$=CHR$(REPLY XOR 32) 'CONVERT TO CAPS
0D54 10EC IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$="?"
0D80 10EC RETURN
0D83 10EC
0D83 10EC 5000 REM $INCLUDE: 'RACS5000.SUB' INCLUDE THE DATE EDITOR SUB

```

```

Offset  Data   Source Line
-----  -
0084  10EC  REM  *****
0084  10EC  REM  ****  AMBULATORY CARE DATA BASE          13 APR 85  ****
0084  10EC  REM  ****  SKIP COLE  ****
0084  10EC  REM  ****  SUBROUTINE NAME      :  RXXS5000.SUB  ****
0084  10EC  REM  ****  SCANNER PROGRAM #    :  ALL  ****
0084  10EC  REM  ****  FUNCTION              :  THIS SUBROUTINE MODULE  ****
0084  10EC  REM  ****  PERFORMS A DATE EDIT  ****
0084  10EC  REM  ****
0084  10EC  REM  ****  INPUT                :  DATE TO BE CHECKED MUST BE  ****
0084  10EC  REM  ****  IN THE VARIABLE NAMED  ****
0084  10EC  REM  ****  'CK.5000$'  ****
0084  10EC  REM  ****  IN THE FORMAT "YYMMDD"  ****
0084  10EC  REM  ****
0084  10EC  REM  ****  OUTPUT              :  'RT.5000' IS THE RETURN CODE  ****
0084  10EC  REM  ****  VARIABLE. IF THIS VARIABLE  ****
0084  10EC  REM  ****  CONTAINS ANY NUMBER OTHER  ****
0084  10EC  REM  ****  THAN 0, AN ERROR WAS FOUND  ****
0084  10EC  REM  ****  IN THE DATE.  ****
0084  10EC  REM  ****
0084  10EC  REM  ****  RESERVED LINE  ****
0084  10EC  REM  ****  NUMBERS          :  5001-5009  ****
0084  10EC  REM  *****
0084  10EC  RT.5000 = 0
0088  10EC  CKYEAR = VAL(LEFT$(CK.5000$,2))  'YEAR NUMERIC VALUE
009E  10EE  CKMONTH = VAL(MID$(CK.5000$,3,2))  'MONTH NUMERIC VALUE
0084  10F0  CKDAY = VAL(RIGHT$(CK.5000$,2))  'DAY NUMERIC VALUE
00C7  10F2
00C7  10F2  IF CKMONTH < 1 THEN RT.5000=1 : GOTO 5009
00DD  10F2  IF CKMONTH > 12 THEN RT.5000=1 : GOTO 5009
00F3  10F2  IF CKDAY < 1 THEN RT.5000=2 : GOTO 5009
00E9  10F2  IF CKDAY > 31 THEN RT.5000=2 : GOTO 5009
0E1F  10F2
0E1F  10F2  REM  LEAP YEAR CHECK
0E1F  10F2  MOLENGTH(2) = 28
0E26  10F2  IF CKMONTH<> 2 THEN GOTO 5005  'MUST BE FEBRUARY
0E35  10F2  IF (CKYEAR MOD 4) <> 0 THEN GOTO 5005  'MUST BE A LEAP YEAR
0E4A  10F2  MOLENGTH(2) = 29
0E51  10F2
0E51  10F2  5005  IF CKDAY > MOLENGTH(CKMONTH) THEN RT.5000=3 : GOTO 5009
0E70  10F2
0E70  10F2  5009  RETURN
0E73  10F2
0E73  10F2  REM  -----END OF SUBROUTINE 5000 -----
0E73  10F2
0E73  10F2  5010  REM $INCLUDE: 'RACS5010.SUB'  INCLUDE THE NUMERIC STRING EDITOR
0E74  10F2  REM  *****
0E74  10F2  REM  ****  AMBULATORY CARE DATA BASE          1 MAY 85  ****
0E74  10F2  REM  ****  SKIP COLE  ****
0E74  10F2  REM  ****  SUBROUTINE NAME      :  RXXS5010.SUB  ****
0E74  10F2  REM  ****  SCANNER PROGRAM #    :  ALL  ****
0E74  10F2  REM  ****  FUNCTION              :  THIS SUBROUTINE MODULE  ****
0E74  10F2  REM  ****  PERFORMS A NUMERIC STRING  ****
0E74  10F2  REM  ****  EDIT.  ****
0E74  10F2  REM  ****

```

Offset	Data	Source Line
0E74	10F2	REM **** INPUT : STRING TO BE EDITED IS IN ****
0E74	10F2	REM **** THE VARIABLE NAMED ****
0E74	10F2	REM **** 'CK.5010\$' ****
0E74	10F2	REM ****
0E74	10F2	REM **** OUTPUT : 'RT.5010' IS THE RETURN CODE ****
0E74	10F2	REM **** VARIABLE. IF THIS VARIABLE ****
0E74	10F2	REM **** CONTAINS ANY NUMBER OTHER ****
0E74	10F2	REM **** THAN 0, AN ERROR WAS FOUND ****
0E74	10F2	REM **** IN THE STRING. ****
0E74	10F2	REM ****
0E74	10F2	REM **** RESERVED LINE ****
0E74	10F2	REM **** NUMBERS : 5011-5019 ****
0E74	10F2	REM ****
0E74	10F2	REM RT.5010 = 0
0E7B	10F2	
0E7B	10F2	FOR I5010 = 1 TO LEN(CK.5010\$)
0E8B	10F4	J5010= ASC(MID\$(CK.5010\$,I5010,1))
0E9F	10F8	IF J5010 < 48 OR J5010 > 57 THEN RT.5010 = RT.5010 + 1
0EC7	10F8	NEXT I5010
0ED8	10F8	RETURN
0ED8	10F8	REM ----- END OF SUBROUTINE 5010 -----
0EDB	10F8	5500 REM INCLUDE: 'RACS5500.SUB' INCLUDE THE OUTPATIENT UCA CHECK SUB
0EDC	10F8	
0EDC	10F8	5700 REM \$INCLUDE: 'RACS5700.SUB' INCLUDE THE MAP ONES TO POSITION NO
0EDD	10F8	REM ****
0EDD	10F8	REM **** AMBULATORY CARE DATA BASE 29 JUL 85 ****
0EDD	10F8	REM **** D R BOLLING ****
0EDD	10F8	REM **** SUBROUTINE NAME : RXXS5700.SUB ****
0EDD	10F8	REM **** SCANNER PROGRAM # : ALL ****
0EDD	10F8	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0EDD	10F8	REM **** CONVERTS A BINARY ARRAY INTO ****
0EDD	10F8	REM **** TWO CHAR CODES. ****
0EDD	10F8	REM ****
0EDD	10F8	REM **** INPUT : INP.STO\$ AS STRING ****
0EDD	10F8	REM ****
0EDD	10F8	REM ****
0EDD	10F8	REM **** OUTPUT : BUF.STO\$ AS STRING. ****
0EDD	10F8	REM ****
0EDD	10F8	REM ****
0EDD	10F8	REM **** RESERVED LINE ****
0EDD	10F8	REM **** NUMBERS : 5710-5730 ****
0F7D	10F8	REM ****
0F8D	10F8	
0EDD	10F8	BUF.STO\$=""
0EE6	10F8	N.STO=1
0EF0	10FA	5710 X.STO=INSTR(N.STO,INP.STO\$,"1")
0EF7	10FC	IF X.STO=0 THEN GOTO 5720 'THATS ALL
0F0E	10FC	N.STO = X.STO + 1 'NEXT STARTING POINT
0F16	10FC	X.STO = X.STO + 100 'PAD WITH LEADING ZERO
0F20	10FC	BUF.STO\$ = BUF.STO\$ + RIGHT\$(STR\$(X.STO),2)
0F36	10FC	IF N.STO <= LEN(INP.STO\$) THEN GOTO 5710
0F49	10FC	
0F49	10FC	5720 RETURN

Offset	Data	Source Line
0F4C	10FC	
0F4C	10FC	REM ----- END OF SUBROUTINE 5700 -----
0F4C	10FC	6000 REM \$INCLUDE: 'RACS6000.SUB' INCLUDE THE INSTRING DECODE SUB
0F4D	10FC	REM *****
0F4D	10FC	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0F4D	10FC	REM **** SKIP COLE ****
0F4D	10FC	REM **** SUBROUTINE NAME : RXXS6000.SUB ****
0F4D	10FC	REM **** SCANNER PROGRAM # : ALL ****
0F4D	10FC	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0F4D	10FC	REM **** PERFORMS INSTRING SEARCH ****
0F4D	10FC	REM ****
0F4D	10FC	REM **** INPUT : STRING TO BE SEARCHED MUST ****
0F4D	10FC	REM **** BE IN THE VARIABLE NAMED : ****
0F4D	10FC	REM **** 'XS' ****
0F4D	10FC	REM ****
0F4D	10FC	REM **** OUTPUT : 'TOT' = TOTAL NUMBER OF ****
0F4D	10FC	REM **** HITS IN THE DESTING ****
0F4D	10FC	REM **** 'HOLD\$()' IS THE ARRAY ****
0F4D	10FC	REM **** CONTAINING THE NUMERIC ****
0F4D	10FC	REM **** VALUE OF THE HIT POSITIONS ****
0F4D	10FC	REM ****
0F4D	10FC	REM *****
0F4D	10FC	REM **** RESERVED LINE ****
0F4D	10FC	REM **** NUMBERS : 6001-6009 ****
0F4D	10FC	REM *****
0F4D	10FC	6001 PTR = INSTR(X\$, "1")
0F58	10FC	TOT = 0
0F62	10FC	WHILE PTR > 0
0F6D	10FC	TOT=TOT+1
0F75	10FC	HOLD\$(TOT) = RIGHT\$(STR\$(PTR),2)
0F97	10FC	PTR=PTR+1
0F9F	10FC	PTR = INSTR(PTR,X\$, "1")
0FB1	10FC	WEND
0FB5	10FC	RETURN
0FB8	10FC	
0FB8	10FC	REM -----END OF SUBROUTINE RXXS6000.SUB-----
0FB8	10FC	
0FB8	10FC	
0FB8	10FC	7000 REM \$INCLUDE: 'RACS7000.SUB' INCLUDE THE SCREEN HEADER SUB
0FB9	10FC	REM *****
0FB9	10FC	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0FB9	10FC	REM **** SKIP COLE ****
0FB9	10FC	REM **** SUBROUTINE NAME : RACS7000.SUB ****
0FB9	10FC	REM **** SCANNER PROGRAM # : ALL ****
0FB9	10FC	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0FB9	10FC	REM **** PRINTS THE STANDARD SCREEN ****
0FB9	10FC	REM **** HEADING. ****
0FB9	10FC	REM **** INPUT : COMMON VARIABLE USERS(2) ****
0FB9	10FC	REM **** SYSTEM DATE ****
0FB9	10FC	REM ****
0FB9	10FC	REM **** OUTPUT : SCREEN HEADING ****
0FB9	10FC	REM ****
0FB9	10FC	REM **** RESERVED LINE ****
0FB9	10FC	REM **** NUMBERS : 7001-7010 ****

Offset	Data	Source Line
0FB9	10FC	REM *****
0FB9	10FC	
0FB9	10FC	7001 LOCATE 1,1
0FC3	10FC	PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
0FCB	10FC	LOCATE 1,65
0FD8	10FC	PRINT DATES;
0FE0	10FC	LOCATE 2,1
0FED	10FC	PRINT "USER : ";USERS(1)
0FFA	10FC	RETURN
0FFD	10FC	7100 REM \$INCLUDE: 'RACS7100.SUB' INCLUDE THE PRINTER HEADER SUB
0FFE	10FC	REM *****
0FFE	10FC	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0FFE	10FC	REM **** SKIP COLE ****
0FFE	10FC	REM **** SUBROUTINE NAME : RXXS7100.SUB ****
0FFE	10FC	REM **** SCANNER PROGRAM # : ALL ****
0FFE	10FC	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0FFE	10FC	REM **** PRINTS THE STANDARD HEADING ****
0FFE	10FC	REM **** ON THE PRINTER. ****
0FFE	10FC	REM **** INPUT : DATE,PAGE,PGMID\$,PGMTITL\$ ****
0FFE	10FC	REM **** ****
0FFE	10FC	REM **** OUTPUT : PRINTER HEADING, LN.COUNT ****
0FFE	10FC	REM **** ****
0FFE	10FC	REM **** RESERVED LINE ****
0FFE	10FC	REM **** NUMBERS : 7101-7110 ****
0FFE	10FC	REM *****
0FFE	10FC	
0FFE	10FC	7101 IF PAGE > 0 THEN LPRINT CHR\$(12);
1014	10FC	LPRINT "ARMY AMBULATORY CARE INFORMATION SYSTEM.... ";PGMTITL\$;
1021	10FC	LPRINT TAB(70);DATES
1034	10FC	PAGE=PAGE+1
103C	10FC	LPRINT "PROGRAM ";PGMID\$;TAB(70);"PAGE";
1059	10FC	LPRINT USING "####";PAGE
1065	10FC	LPRINT
1060	10FC	LN.COUNT=3
1074	10FC	RETURN
1077	10FC	
1077	10FC	8000 REM \$INCLUDE: 'RACS8000.SUB' INCLUDE THE DECODE SUB GROUP
1078	10FC	REM *****
1078	10FC	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ***
1078	10FC	REM **** SKIP COLE ***
1078	10FC	REM **** SUBROUTINE NAME : RXXS8000.SUB ***
1078	10FC	REM **** SCANNER PROGRAM # : ALL ***
1078	10FC	REM **** FUNCTION : THIS SUBROUTINE MODULE ***
1078	10FC	REM **** IS A GROUPING THAT PERFORMS ***
1078	10FC	REM **** VARIOUS DECODING FUNCTIONS ***
1078	10FC	REM **** ON THE SCANNER DATA ***
1078	10FC	REM **** ****
1078	10FC	REM **** 8001 - DECODE THE HEADER POSITIONS (POINTER 0-20) ***
1078	10FC	REM **** 8050 - CHECK FOR END OF JOB ***
1079	10FC	REM **** 8100 - PRINT THE HEADER DATA ON THE SCREEN ***
1078	10FC	REM **** 8200 - DECODE THE RESPONSE POSITIONS (POINTER 21-...) ***
1078	10FC	REM **** (RETURNED IN TEXT\$ STRING VARIABLE) ***
1078	10FC	REM **** ****
1078	10FC	REM **** INPUT : SHEET RECORD, RECORD LENGTH ***

Offset	Data	Source Line
1078	10FC	REM **** ***
1078	10FC	REM **** OUTPUT : 'TEXTS' TRING VARIABLE ***
1078	10FC	REM **** ***
1078	10FC	REM **** RESERVED LINE ***
1078	10FC	REM **** NUMBERS : 8001-8500 ***
1078	10FC	REM *****
1078	10FC	
1078	10FC	'DECODE THE HEADER ONLY
1078	10FC	8001 POINTER = 0
107F	10FC	RECORDPTR = VARPTR(SHEETREC(0))
1086	10FE	FOR J8000 = 1 TO 21
1080	10FE	8002 TEXTS= TEXTS+CHR\$(PEEK(RECORDPTR + POINTER))
10A8	10FE	POINTER=POINTER+1
10B3	10FE	NEXT J8000
10C2	1100	PROGRAMS= LEFT\$(TEXT\$,3)
10D1	1104	BATCHS= MID\$(TEXT\$,4,3)
10E3	1108	SERIALS= MID\$(TEXT\$,7,4)
10F5	110C	RUNIDS= MID\$(TEXT\$,11,1)
1107	1110	FORMS= MID\$(TEXT\$,12,2)
1119	1114	POCKETS= MID\$(TEXT\$,14,1)
112B	1118	SCANERR1S=MID\$(TEXT\$,16,2)
1130	111C	SCANERR2S=MID\$(TEXT\$,18,2)
114F	1120	SCANERR3S=MID\$(TEXT\$,20,2)
1161	1124	GOTO 8500
1165	1124	
1165	1124	8050 REM CHECK FOR END OF JOB/END OF BATCH
1166	1124	IF PROGRAMS = PGMID\$ THEN GOTO 8500
1178	1124	LPRINT STRING\$(80,"")
1186	1124	LPRINT
118E	1124	LPRINT "RECORDS PROCESSED ... ";SERIALS
119B	1124	LPRINT "STARTED AT ";BTIMES
11A8	1124	LPRINT "ENDED AT ";TIMES
11B5	1124	LPRINT CHR\$(12)
11C0	1124	GOTO 30000
11C4	1124	
11C4	1124	8070 REM CHECK FOR SCANNER ERRORS
11C5	1124	IF POCKETS = " " GOTO 8500
11D7	1124	LPRINT LITHOS;
11DF	1124	LPRINT " ... SCANNER ERRORS : ";
11E7	1124	LPRINT SCANERR1\$;" / ";
11F4	1124	LPRINT SCANERR2\$;" / ";
1201	1124	LPRINT SCANERR3\$
1209	1124	LN=LN+1
1211	1126	GOTO 999
1215	1126	
1215	1126	8100 REM PRINT THE HEADER VARIABLES ON THE TUBE....
1216	1126	LOCATE 5,1:PRINT "PROGRAM ";PROGRAMS;
1230	1126	
1230	1126	PRINT " BATCH ";BATCHS;
1230	1126	PRINT " RUN ";RUNIDS;
124A	1126	PRINT " FORM ";FORMS;
1257	1126	PRINT " POCKET ";POCKETS
1264	1126	GOTO 8500
1268	1126	

```

Offset  Data  Source Line
1268 1126 8200 REM DECODE THE RESPONSE POSITIONS
1269 1126      POINTER = 21
1270 1126      RECORDPTR = VARPTR(SHEETREC(0))
1277 1126      FOR J8000 = 22 TO RECORDLENGTH
1284 1128 8202      TEXT$ = TEXT$+CHR$(PEEK(RECORDPTR + POINTER))
12A2 1128      POINTER=POINTER+1
12AA 1128      NEXT J8000
12BB 1128
12BB 1128 8500 RETURN
12BE 1128
12BE 1128 REM ----- END OF RXXS8000.SUB -----
12BE 1128
12BE 1128 9000 REM $INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUB
12BF 1128 REM *****
12BF 1128 REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
12BF 1128 REM **** SKIP COLE ****
12BF 1128 REM **** PROGRAM NAME : RACS9000.SUB ****
12BF 1128 REM **** SCANNER PROGRAM # : ALL ****
12BF 1128 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
12BF 1128 REM **** CONTROLS THE SCANNER I/O ****
12BF 1128 REM ****
12BF 1128 REM **** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS ****
12BF 1128 REM **** COMMUNICATIONS MANUAL AND THE ****
12BF 1128 REM **** PRE-RELEASED SOFTWARE GUIDE ****
12BF 1128 REM ****
12BF 1128 REM *****
12BF 1128 REM **** RESERVED LINE ****
12BF 1128 REM **** NUMBERS : 9001-9100 ****
12BF 1128 REM *****
12BF 1128
12BF 1128 REM *****
12BF 1128 REM **** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
12BF 1128 REM **** ARGUMENTS: PRESET ... SEE BELOW ****
12BF 1128 REM *****
12BF 1128 9001 REM
12C0 1128      PROTOCOL(0) = 9600 'BAUD RATE
12C7 1128      PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
12CE 1128      PROTOCOL(2) = 8 'DATA BITS
12D5 1128      PROTOCOL(3) = 1 'STOP BITS
12D0 1128      PROTOCOL(4) = 2 'RS-232 PORT
12E3 1128      PROTOCOL(5) = 0 'WRITE TIME-OUT
12EA 1128      PROTOCOL(6) = 0 'READ TIME-OUT
12F1 1128
12F1 1128      ERRSTAT$ = SPACE$(60)
12FD 1128      ARGPTR = VARPTR(PROTOCOL(0))
1304 112A      CALL SETUP (ARGPTR,ERRSTAT$)
1315 112A      ERRMSG$=""
131E 112A      IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="SETUP ERROR "+ERRSTAT$
133A 112A      GOTO 9100
133E 112A
133E 112A REM *****
133E 112A REM **** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
133E 112A REM **** ARGUMENTS: CNTRLOPT ****

```

```

133E 112A REM **** CNTRLOPT = 1 = START SCANNER (S1) ****
133E 112A REM **** CNTRLOPT = 2 = STOP SCANNER (SO) ****
133E 112A REM **** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
133E 112A REM **** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
133E 112A REM **** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
133E 112A REM **** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
133E 112A REM **** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
133E 112A REM **** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****
133E 112A REM *****
133E 112A 9010 REM
133F 112A ERRSTAT$ = SPACES(60)
134B 112A CALL CNTRLOP (CNTRLOPT,ERRSTAT$)
135C 112A ERRMSG$=""
1365 112A IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="CONTROL ERROR "+ERRSTAT$
1381 112A GOTO 9100
1385 112A
1385 112A REM *****
1385 112A REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
1385 112A REM ****
1385 112A REM **** ARGUMENTS: READTYPE ****
1385 112A REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
1385 112A REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
1385 112A REM ****
1385 112A REM **** ARGUMENTS: RECORDLENGTH ****
1385 112A REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
1385 112A REM **** TRANSMITTED ****
1385 112A REM *****
1385 112A 9020 REM
1386 112A ERRSTAT$ = SPACES(60)
1392 112A RECORDPTR = VARPTR(SHEETREC(0))
1399 112A CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT$)
13B2 112A ERRMSG$=""
13B8 112A IF MID$(ERRSTAT$,14,3) = "415" THEN ERRMSG$="ESC"
13DC 112A GOTO 9100
13E0 112A
13E0 112A REM *****
13E0 112A REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
13E0 112A REM ****
13E0 112A REM **** ARGUMENTS: PRINTPOS ****
13E0 112A REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
13E0 112A REM **** VALUES = 0 THRU 90 ****
13E0 112A REM ****
13E0 112A REM **** ARGUMENTS: PSTRINGS ****
13E0 112A REM **** TEXT TO BE PRINTED ON THE FORM ****
13E0 112A REM ****
13E0 112A REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
13E0 112A REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
13E0 112A REM *****
13E0 112A 9030 REM
13E1 112A ERRSTAT$ = SPACES(60)
13ED 112A RECORDPTR = VARPTR(SHEETREC(0))
13F4 112A CALL TPRINT(PRINTPOS,PSTRINGS,ERRSTAT$)
1409 1130 IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="PRINT ERROR "+ERRSTAT$
1425 1130 GOTO 9100

```

RACP950
SOCIAL WORK SHORT FORM

PAGE 20
07-06-87
15:11:33

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```
1429 1130
1429 1130 9100 RETURN
142C 1130 REM -----END OF SUBROUTINE RACS9000.SUB -----
142C 1130
142C 1130 REM END OF SUBROUTINES =====
142C 1130
142C 1130 25000 REM USER TERMINATED INPUT... FILE IS NOT TO BE USED!!!!
142D 1130 LPRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1435 1130 LPRINT "ERASING FILE ";DATFIL$
1442 1130 BEEP
1446 1130 CLS : PRINT "USER TERMINATED INPUT .. DATA WILL NOT BE USED!"
1452 1130 CLOSE
1456 1130 OPEN DATFIL$ FOR OUTPUT AS #1
1468 1130 PRINT #1,STRING$(RECORDLENGTH,"X") 'VOID THE FIRST RECORD
147A 1130 CLOSE
147E 1130
147E 1130 30000 REM
147F 1130 CLOSE
1483 1130 CHAIN "RACP10"
148A 1130 END
148E 1130
1491 1130
```

22151 Bytes Available
16932 Bytes Free

0 Warning Error(s)
0 Severe Error(s)

APPENDIX C

BASIC SYSTEM PROGRAMS (REFERENCED SECTION 3.4.4, 5.2.2)

APPENDIX C

PC BASIC SYSTEM AND MISC. PROGRAMS

TABLE OF CONTENTS

RACP05.LST	Initial Sign-On
RACP10.LST	Initial Selection Menu
RACP1A.LST	OMR (Scanner) Select
CONVERT	SIDPERS Conversion

Offset	Data	Source Line
--------	------	-------------

001A	0002	REM \$LINESIZE :132
001A	0002	REM \$PAGESIZE: 66
001A	0002	REM \$TITLE: 'RACPO5'
001A	0002	REM \$SUBTITLE: 'Initial Sign-on screen'
001A	0002	REM \$PAGE

Offset Data Source Line IBM Personal Computer BASIC Compiler V1.00

```

001A 0002 REM +-----+
001A 0002 REM | NAME: RACP05      AMBULATORY CARE INFORMATION SYSTEM LOGON |
001A 0002 REM |                               WRITTEN BY: FLOYD COLE |
001A 0002 REM | MODIFIED FOR NEW FORM DESIGN  2 JAN 87 BY DAVID BOLLING |
001A 0002 REM +-----+
001A 0002 REM This is the first program in the Ambulatory care series. Its
001A 0002 REM purpose is to log the user on to the system. From there,
001A 0002 REM control is passed to RACP10 and a MAIN menu which allows the user
001A 0002 REM to invoke various system options.
001A 0002 REM
001A 0002 REM The program is called from the autoexec.bat file on boot.
001A 0002 REM Control is returned to this program on logoff from any of the
001A 0002 REM other basic system modules. The system is a closed, tightly
001A 0002 REM controlled loop.
001A 0002 REM -----
001A 0002 REM $INCLUDE: 'RACDIM.MOD'      REM Include the DIMENSION DEFINITIONS
001A 0002 REM *****
001A 0002 '* NAME: RACDIM.MOD              DIMENSION DEFINITIONS *
001A 0002 '* Date: 28 Feb 84              Written by: Floyd Cole *
001A 0002 REM *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM $INCLUDE: 'RACCMN.MOD'      REM Include the COMMON AREA DEFINITION
001A 0002 REM *****
001A 0002 '* NAME: RACCMN.MOD              COMMON AREA DEFINITION *
001A 0002 '* Date: 28 Feb 84              Written by: Floyd Cole *
001A 0002 REM *****
001A 0002 ' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A 0002 ' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re*compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 '
001A 0002 ' *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002 COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL$ 'BASIC SCREEN COLORS
001A 0002 COMMON HEADERS$ '21 CHARACTER SCANNER HEADER INFO
001A 0002 COMMON TEXT$ ' AINING CHARACTERS FROM SCANNER
001A 0002 COMMON PGMIDS$ 'PROGRAM OR FORM ID
001A 0002 COMMON MOLENGTH() 'DAYS IN THE MONTH
001A 0002 COMMON USERS()

```

Jffset Data Source Line

```

001A 0002 ' *****END OF COMMON DEFINITION*****
001A 0002
001A 0002 REM $INCLUDE: 'RACFUN.MOD' REM Include the FUNCTION DEFINITIONS
001A 0002 REM +-----+
001A 0002 REM | NAME: RXXFUN.MOD FUNCTIONS DEFINITIONS |
001A 0002 REM | Date: 28 Feb 84 Written by: Floyd Cole |
001A 0002 REM +-----+
001A 0002 REM Defined functions will held in this file.
001A 0002 REM It is an included file so it cannot be run in a stand-alone,
001A 0002 REM mode.
001A 0002 REM
001A 0002 REM This program segment may be modified, but all files containing
001A 0002 REM an include for this segment must be re-compiled in order to
001A 0002 REM affect the changes made here.
001A 0002 REM
001A 0002 REM -----START OF FUNCTION DEFINITION-----
001A 0002 REM NO FUNCTIONS CURRENTLY DEFINED
001A 0002 ON KEY (1) GOSUB 3001
0040 0012 ON KEY (2) GOSUB 3002
004A 0012 ON KEY (3) GOSUB 3003
0054 0012 ON KEY (4) GOSUB 3004
005E 0012 ON KEY (5) GOSUB 3005
0068 0012 ON KEY (6) GOSUB 3006
0072 0012 ON KEY (7) GOSUB 3007
007C 0012 ON KEY (8) GOSUB 3008
0086 0012 ON KEY (9) GOSUB 3009
0090 0012 ON KEY (10) GOSUB 3010
009A 0012 REM -----END OF FUNCTION DEFINITION-----
009A 0012 REM $INCLUDE: 'RACDEF.MOD' REM Include the DEFAULT DEFINITIONS
009A 0012 *****
009A 0012 '* NAME: RACP01.DEF DEFAULT DEFINITIONS *
009A 0012 '* Date: 28 Feb 84 Written by: Floyd Cole *
009A 0012 *****
009A 0012 ' Variables used in common that have a default value on start*up
009A 0012 ' will be held in this file. It is an included file so it cannot
009A 0012 ' be run in a stand*alone mode. In normal operation, this file
009A 0012 ' should be 'included' in the main program only (RACP10.BAS).
009A 0012 '
009A 0012 ' This program segment may be modified, but all files containing
009A 0012 ' an include for this segment must be re*compiled in order to
009A 0012 ' affect the changes made here.
009A 0012 '
009A 0012 ' *****START OF DEFAULT DEFINITION*****
009A 0012 FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
00A1 0012 BACK = 1 'Background Color = Light Blue
00A8 0012 BORD = 4 'BORDER = RED
00AF 0014 HIDE = 4 'ALTERNATE COLOR = RED
00B6 0014 EFORE= 14 'ERROR FOREGROUND DISPLAY
00B8 0014 EBACK= 0 'ERROR BACKGROUND DISPLAY
00C4 0014 BELL$ = CHR$(7) 'Sound the bell
00D0 0014
00D0 0014 MOLENGTH(1) = 31 'JAN

```

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```

0007 0014      MOLENGTH(2) = 28      'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
000E 0014      MOLENGTH(3) = 31      'MAR
00E5 0014      MOLENGTH(4) = 30      'APR
00EC 0014      MOLENGTH(5) = 31      'MAY
00F3 0014      MOLENGTH(6) = 30      'JUN
00FA 0014      MOLENGTH(7) = 31      'JUL
0101 0014      MOLENGTH(8) = 31      'AUG
0108 0014      MOLENGTH(9) = 30      'SEP
010F 0014      MOLENGTH(10) = 31     'OCT
0116 0014      MOLENGTH(11) = 30     'NOV
011D 0014      MOLENGTH(12) = 31     'DEC
0124 0014
0124 0014      DATEERR$(0) = " "
012D 0014      DATEERR$(1) = "INVALID MONTH"
0136 0014      DATEERR$(2) = "INVALID DAY "
013F 0014      DATEERR$(3) = "DAY TOO LARGE FOR MONTH CODED"
0148 0014
0148 0014      MAXLENGTH = 80        'MAXIMUM LENGTH OF OUTPUT RECORD
014F 0016      PAD$ = " "          'PAD CHARACTER FOR SHORT RECORDS
0158 001A
0158 001A      ' *****END OF DEFAULT DEFINITION*****
0158 001A
0158 001A      REM -----
0158 001A      10 KEY (3) ON
015F 001A      COLOR FORE,BACK,BORD
0175 001A      CLOSE
0179 001A      USER$(1)="*****"
0182 001A      USER$(2)="*****"
0188 001A      100 CLS
018F 001A      110 GOSUB 7000      'PRINT THE HEADING
0194 001A
0194 001A      OPEN "O",1,"RACB00.BAT"
01A6 001A      PRINT #1, "CLS"
01B1 001A      CLOSE
01B5 001A
01B5 001A      OPEN "O",1,"RACLOG.DAT"
01C7 001A      CLOSE
01CB 001A
01CB 001A      200 REM SETUP THE LOGON SCREEN
01CC 001A      LOCATE 12,30
01D9 001A      PRINT "PLEASE LOG ON:  "
01E1 001A      LOCATE 16,30
01EE 001A      PRINT "USER CODE ..... "
01F6 001A      LOCATE 18,30
0203 001A      PRINT "PASSWORD ..... "
0208 001A      LOCATE 22,24
0218 001A      PRINT "KEY 'END' IN USER CODE TO EXIT"
0220 001A
0220 001A      300 REM INPUT THE USER CODE AND PASSWORD
0221 001A      LOCATE 16,50      'CLEAR THE PREVIOUS ENTRIES
022E 001A      PRINT SPC(30);
023C 001A      LOCATE 18,50
0249 001A      PRINT SPC(30);
0257 001A

```

RACP05

Initial Sign-on screen

PAGE 5

04-09-87

14:04:51

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```

0257 001A      LOCATE 16,50          'POSITION AND RETRIEVE
0264 001A      LINE INPUT;USER$(1)
0270 001A      IF USER$(1) <> "END" AND USER$(1) <> "end" THEN GOTO 307
0297 001A
0297 001A      OPEN "O",1,"RACFPWD.FEX"
02A9 001A      PRINT #1,"*****"
02B4 001A      CLOSE
02B8 001A
02B8 001A      COLOR 7,0,0
02C7 001A      CLS
02CB 001A      END
02CF 001A
02CF 001A      307 LOCATE 18,50
02DC 001A      COLOR HIDE,HIDE,BORD    'HIDE THE INPUT
02EE 001A      LINE INPUT;USER$(2)
02FA 001A      COLOR FORE,BACK,BORD    'RESET THE COLOR PATTERN
0310 001A
0310 001A      310 REM CHECK FOR THE OBVIOUS
0311 001A      IF USER$(1)=" " OR USER$(2)=" " GOTO 300 'BLANKS
0338 001A      CLS
033C 001A
033C 001A      400 REM CONVERT TO CAPS
033D 001A      FOR J = 1 TO 2          'CONVERT USER.ID AND PASSWORD
0344 001A      A$=USER$(J)
0355 0020      FOR I = 1 TO LEN(A$)
0366 0022      B$=MID$(A$,I,1)
0379 0028      IF ASC(B$) > 47 AND ASC(B$)< 91 THEN 450
03A0 0028      MID$(A$,I,1)=CHR$(ASC(MID$(A$,I,1)) XOR 32)
03BC 0028      450 NEXT I              'END OF CAPS
03CD 0028      USER$(J)=A$           'REPLACE THE VARIABLE
03E1 0028      NEXT J
03F3 0028
03F3 0028      IF USER$(2)="LUGNUT" THEN 470
0401 0028      IF USER$(2)="NUTMEG" THEN 470
040F 0028      IF USER$(2)="REDWAX" THEN 470
041D 0028      IF USER$(2)="WETHEN" THEN 470
042B 0028      IF USER$(2)="BIGJAR" THEN 470
0439 0028      IF USER$(2)="BAYRUM" THEN 470
0447 0028
0447 0028      LOCATE 20,30
0454 0028      PRINT "UNAUTHORIZED PASSWORD"
045C 0028      GOTO 110
0460 0028
0460 0028      470 REM WRITE ENCRYPTED PASSWORD
0461 0028      OPEN "RACFPWD.FEX" FOR OUTPUT AS #2
0473 0028      FOR I=1 TO 6
047A 0028      PWS=PWS+CHR$(ASC(MID$(USER$(2),7-I,1))+10+1)
049F 002C      NEXT I
04AE 002C      PWS=PWS+CHR$(ASC(MID$(USER$(2),4,1))+17)
04CB 002C      PRINT #2,PWS
04D6 002C      CLOSE 2
04DD 002C
04DD 002C      OPEN "O",1,"RACLOG.DAT"
04EF 002C      P1$="RACP05"

```

RACPD5
Initial Sign-on screen

PAGE 6
04-09-87
14:04:51

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```
04F8 0030      WRITE #1, USER$(1),DATE$,TIME$,PID$
0513 0030      CLOSE
0517 0030      CHAIN "RACP10"
051E 0030
051E 0030
051E 0030 3000 REM PF KEY TRAPS
051F 0030
051F 0030 3001 REM PF 1 = HELP
0520 0030      KEY (1) OFF          'DISABLE PF KEY
0527 0030      RETURN 10
052D 0030
052D 0030 3002 REM PF 2 = UNUSED
052E 0030      KEY (2) OFF          'DISABLE PF KEY
0535 0030      RETURN 10
0538 0030
0538 0030 3003 REM PF 3 = UNUSED
053C 0030      CLOSE : CLS : END
0548 0030
0548 0030 3004 REM PF 4 = UNUSED
0549 0030      KEY (4) OFF          'DISABLE PF KEY
0550 0030      RETURN 10
0556 0030
0556 0030 3005 REM PF 5 = UNUSED
0557 0030      KEY (5) OFF          'DISABLE PF KEY
055E 0030      RETURN 10
0564 0030
0564 0030 3006 REM PF 6 = UNUSED
0565 0030      KEY (6) OFF          'DISABLE PF KEY
056C 0030      RETURN 10
0572 0030
0572 0030 3007 REM PF 7 = UNUSED
0573 0030      KEY (7) OFF          'DISABLE PF KEY
057A 0030      RETURN 10
0580 0030
0580 0030
0580 0030 3008 REM PF 8 = UNUSED
0581 0030      KEY (8) OFF          'DISABLE PF KEY
0588 0030      RETURN 10
058E 0030
058E 0030 3009 REM PF 9 = UNUSED
058F 0030      KEY (9) OFF          'DISABLE PF KEY
0596 0030      RETURN 10
059C 0030
059C 0030 3010 REM PF10 = UNUSED
059D 0030      KEY (10) OFF         'DISABLE PF KEY
05A4 0030      RETURN 10
05AA 0030
05AA 0030 7000 REM PRINT THE HEADING ON THE SCREEN
05AB 0030 REM $INCLUDE: 'RACS7000.SUB' INCLUDE THE STANDARD HEADING
05AB 0030 REM *****
05AB 0030 REM ****    AMBULATORY CARE DATA BASE          13 APR 85    ****
05AB 0030 REM ****                                SKIP COLE      ****
05AB 0030 REM ****    SUBROUTINE NAME      :    RACS7000.SUB      ****
05AB 0030 REM ****    SCANNER PROGRAM #    :    ALL                ****
```

```
05AB 0030 REM **** FUNCTION : THIS SUBROUTINE MODULE ****
05AB 0030 REM **** PRINTS THE STANDARD SCREEN ****
05AB 0030 REM **** HEADING. ****
05AB 0030 REM **** INPUT : COMMON VARIABLE USERS$(2) ****
05AB 0030 REM **** SYSTEM DATE ****
05AB 0030 REM ****
05AB 0030 REM **** OUTPUT : SCREEN HEADING ****
05AB 0030 REM ****
05AB 0030 REM **** RESERVED LINE ****
05AB 0030 REM **** NUMBERS : 7001-7010 ****
05AB 0030 REM ****
05AB 0030
05AB 0030 7001 LOCATE 1,1
05B5 0030 PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
05B0 0030 LOCATE 1,65
05CA 0030 PRINT DATES;
05D2 0030 LOCATE 2,1
05DF 0030 PRINT "USER : ";USERS$(1)
05EC 0030 RETURN
05EF 0030
05EF 0030
05F2 0030
```

22151 Bytes Available

20364 Bytes Free

0 Warning Error(s)
0 Severe Error(s)

PAGE 1

08-03-87

07:17:19

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

001A 0002 REM \$LINESIZE:132

001A 0002 REM \$PAGESIZE: 66

001A 0002 REM \$TITLE: 'RACP10 '

001A 0002 REM \$SUBTITLE:'Initial Selection Screen'

001A 0002 REM \$PAGE

```

Offset Data Source Line
001A 0002 REM +-----+
001A 0002 REM | Name: RACP10      Ambulatory Care Information System MAIN |
001A 0002 REM | Date: 28 Feb 84      Written by: Floyd Cole |
001A 0002 REM | Update: 23 Mar 87      Re-written by: D. R. Bolling |
001A 0002 REM +-----+
001A 0002 REM This is the first program in the Ambulatory care series. Its
001A 0002 REM purpose is to log the user on to the system. From there,
001A 0002 REM a main menu is displayed which allows the user to perform
001A 0002 REM the functions of the system.
001A 0002 REM
001A 0002 REM THE PROGRAM IS CALLED FROM THE AUTOEXEC.BAT FILE ON BOOT.
001A 0002 REM CONTROL IS RETURNED TO THIS PROGRAM ON LOGOFF, EXIT FROM
001A 0002 REM the other basic modules and on exit from PC/Focus. It is
001A 0002 REM a closed system.
001A 0002 REM -----
001A 0002 REM $INCLUDE: 'RACDIM.MOD' REM Include the DIMENSION DEFINITIONS
001A 0002 *****
001A 0002 '* NAME: RACDIM.MOD DIMENSION DEFINITIONS *
001A 0002 '* Date: 28 Feb 84 Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' Dimensioned variables are defined in this file.
001A 0002 ' It is an included file so it cannot be run in a stand-alone,
001A 0002 ' mode.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re-compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 ' ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002 DEFINT A-Z
001A 0002 DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002 ' ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002 REM $INCLUDE: 'RACCMN.MOD' REM Include the COMMON AREA DEFINITION
001A 0002 *****
001A 0002 '* NAME: RACCMN.MOD COMMON AREA DEFINITION *
001A 0002 '* Date: 28 Feb 84 Written by: Floyd Cole *
001A 0002 *****
001A 0002 ' COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A 0002 ' INCLUDED FILE SO IT CANNOT BE RUN IN A STAND-ALONE, MODE.
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re*compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 '
001A 0002 ' *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002 COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL$ 'BASIC SCREEN COLORS
001A 0002 COMMON HEADERS$ '21 CHARACTER SCANNER HEADER INFO
001A 0002 COMMON TEXT$ '11 AINING CHARACTERS FROM SCANNER
001A 0002 COMMON PGMID$ 'PROGRAM OR FORM ID
001A 0002 COMMON MOLENGTH() 'DAYS IN THE MONTH
001A 0002 COMMON USERS$()

```

IBM Personal Computer BASIC Compiler V1.00

```

001A 0002 ' *****END OF COMMON DEFINITION*****
001A 0002
001A 0002 REM $INCLUDE: 'RACFUN.MOD' REM Include the FUNCTION DEFINITIONS
001A 0002 REM +-----+
001A 0002 REM | NAME: RXxFUN.MOD FUNCTIONS DEFINITIONS |
001A 0002 REM | Date: 28 Feb 84 Written by: Floyd Cole |
001A 0002 REM +-----+
001A 0002 REM Defined functions will held in this file.
001A 0002 REM It is an included file so it cannot be run in a stand-alone,
001A 0002 REM mode.
001A 0002 REM
001A 0002 REM This program segment may be modified, but all files containing
001A 0002 REM an include for this segment must be re-compiled in order to
001A 0002 REM affect the changes made here.
001A 0002 REM
001A 0002
001A 0002 REM -----START OF FUNCTION DEFINITION-----
001A 0002 REM NO FUNCTIONS CURRENTLY DEFINED
001A 0002 ON KEY (1) GOSUB 3001
0040 0012 ON KEY (2) GOSUB 3002
004A 0012 ON KEY (3) GOSUB 3003
0054 0012 ON KEY (4) GOSUB 3004
005E 0012 ON KEY (5) GOSUB 3005
0068 0012 ON KEY (6) GOSUB 3006
0072 0012 ON KEY (7) GOSUB 3007
007C 0012 ON KEY (8) GOSUB 3008
0086 0012 ON KEY (9) GOSUB 3009
0090 0012 ON KEY (10) GOSUB 3010
009A 0012 REM -----END OF FUNCTION DEFINITION-----
009A 0012 REM $INCLUDE: 'RACDEF.MOD' REM Include the DEFAULT DEFINITIONS
009A 0012 *****
009A 0012 '* NAME: RACP01.DEF DEFAULT DEFINITIONS *
009A 0012 '* Date: 28 Feb 84 Written by: Floyd Cole *
009A 0012 *****
009A 0012 ' Variables used in common that have a default value on start*up
009A 0012 ' will be held in this file. It is an included file so it cannot
009A 0012 ' be run in a stand*alone mode. In normal operation, this file
009A 0012 ' should be 'included' in the main program only (RACP10.BAS).
009A 0012 '
009A 0012 ' This program segment may be modified, but all files containing
009A 0012 ' an include for this segment must be re*compiled in order to
009A 0012 ' affect the changes made here.
009A 0012 '
009A 0012
009A 0012 ' *****START OF DEFAULT DEFINITION*****
009A 0012 FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
00A1 0012 BACK = 1 'Background Color = Light Blue
00A8 0012 BORD = 4 'BORDER = RED
00AF 0014 HIDE = 4 'ALTERNATE COLOR = RED
00B6 0014 EFORE= 14 'ERROR FOREGROUND DISPLAY
00BD 0014 EBACK= 0 'ERROR BACKGROUND DISPLAY
00C4 0014 BELL$ = CHR$(7) 'Sound the bell
00D0 0014
00D0 0014 MOLENGTH(1) = 31 'JAN

```

Offset	Data	Source Line
0007	0014	MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
000E	0014	MOLENGTH(3) = 31 'MAR
00E5	0014	MOLENGTH(4) = 30 'APR
00EC	0014	MOLENGTH(5) = 31 'MAY
00F3	0014	MOLENGTH(6) = 30 'JUN
00FA	0014	MOLENGTH(7) = 31 'JUL
0101	0014	MOLENGTH(8) = 31 'AUG
0108	0014	MOLENGTH(9) = 30 'SEP
010F	0014	MOLENGTH(10) = 31 'OCT
0116	0014	MOLENGTH(11) = 30 'NOV
011D	0014	MOLENGTH(12) = 31 'DEC
0124	0014	
0124	0014	DATEERR\$(0) = " "
012D	0014	DATEERR\$(1) = "INVALID MONTH"
0136	0014	DATEERR\$(2) = "INVALID DAY "
013F	0014	DATEERR\$(3) = "DAY TOO LARGE FOR MONTH CODED"
0148	0014	
0148	0014	MAXLENGTH = 80 'MAXIMUM LENGTH OF OUTPUT RECORD
014F	0016	PADS = "." 'PAD CHARACTER FOR SHORT RECORDS
0158	001A	
0158	001A	*****END OF DEFAULT DEFINITION*****
0158	001A	
0158	001A	REM -----
0158	001A	
0158	001A	REM CHECK FOR ENTRY
0158	001A	OPEN "1",1,"RACLOG.DAT"
016A	001A	IF EOF(1) THEN 3010 'MAKE THEM LOG ON FIRST
0178	001A	INPUT #1,USERS(1),DT\$,TMS\$,PIDS
0199	0026	IF USERS(1) = "" THEN 3010 'MAKE THEM LOG ON FIRST
01A7	0026	IF USERS(1) = "*****" THEN 3010 'MAKE THEM LOG ON FIRST
01B5	0026	CLOSE 1
018C	0026	
018C	0026	REM READ PASSWORD AND DECRYPT
018C	0026	OPEN "1",1,"RACFPWD.FEX"
01CE	0026	IF EOF(1) THEN 3010
01DC	0026	INPUT #1,PW\$
01EB	002A	CLOSE 1
01F2	002A	USERS(2)=" "
01FB	002A	FOR I=1 TO 6
0202	002A	USERS(2)=USERS(2)+CHR\$(ASC(MID\$(PW\$,7-I,1))-17+I)
0227	002C	NEXT I
0236	002C	
0236	002C	KEY (1) ON
023D	002C	KEY (10) ON
0244	002C	SCREEN 0,1,0,0
025A	002C	COLOR FORE,BACK,BORD
0270	002C	CLS
0274	002C	REM PAGE 2 - THE PLEASE WAIT SCREEN
0274	002C	SCREEN ,,2
0281	002C	CLS
0285	002C	GOSUB 7000 'PRINT HEADING
028A	002C	LOCATE 12,25
0297	002C	PRINT "PLEASE WAIT ... PROGRAM LOADING"
029F	002C	

Offset Data Source Line

```

029F 002C REM
029F 002C REM PUT THE MAIN MENU ON PAGE 0 WHILE DISPLAYING PAGE 2
029F 002C SCREEN ,,0,2
02B1 002C CLS
02B5 002C GOSUB 7000 'PRINT HEADING
02BA 002C REM $INCLUDE: 'RACS10.M01' 'INCLUDE THE MAIN MENU
02BA 002C REM +-----+
02BA 002C REM | NAME: RACS10.M01 RACS10: MAIN MENU |
02BA 002C REM | DATE: 27 AUG 86 WRITTEN BY: D R BOLLING |
02BA 002C REM +-----+
02BA 002C REM This screen is the main menu used in the opening program.
02BA 002C REM Macro level functions are displayed and selected from this screen.
02BA 002C REM
02BA 002C REM REMINDER: The first two lines of display are reserved for the
02BA 002C REM heading, and the last three lines are reserved for selection input,
02BA 002C REM error display and function display.
02BA 002C REM
02BA 002C REM User menus are restricted to lines 3-21.
02BA 002C REM
02BA 002C REM -----START OF MAIN MENU SELECTION-----
02BA 002C ROW=3:COL=1 'Starting ROW and COLUMN for menu display
02C8 0030 LOCATE ROW,COL
02D7 0030 PRINT "
02DF 0030 PRINT "
02E7 0030 PRINT "
02EF 0030 PRINT "
02F7 0030 PRINT "
02FF 0030 PRINT "
0307 0030 PRINT "
030F 0030 PRINT "
0317 0030 PRINT "
031F 0030 PRINT "
0327 0030 PRINT "
032F 0030 PRINT "
0337 0030 PRINT "
033F 0030 PRINT "
0347 0030 PRINT "
034F 0030 PRINT "
0357 0030
0357 0030 LOCATE 20,1
0364 0030 PRINT "
036C 0030 LOCATE 20,40
0379 0030 REM -----END OF MAIN MENU SELECTION-----
0379 0030
0379 0030 REM UPDATE THE FILE
0379 0030 OPEN "O",1,"RACLOG.DAT"
0388 0030 PIDS="RACP10"
0394 0030 WRITE #1,USERS(1),DATES,TIMES,PIDS
03AF 0030 CLOSE 1
0386 0030
0386 0030 600 REM OPEN THE COMMON BATCH FILE AND INITIALIZE IT
0387 0030 OPEN "O",1,"RACB00.BAT"
03C9 0030

```

MAIN MENU

Option	Description	Option	Description
A ...	START OPTICAL MARK READER	F ...	FUTURE APPLICATION
B ...	COPY VISIT.DAT TO D DRIVE	G ...	FUTURE APPLICATION
C ...	COPY PATIENT.DAT TO D	H ...	FUTURE APPLICATION
D ...	COPY PROVIDER.DAT TO D	I ...	FUTURE APPLICATION
E ...	FUTURE APPLICATION	X ...	EXIT

Enter Option →

Initial Selection Screen

08-03-87

07:17:19

IBM Personal Computer BASIC Compiler V1.00

Offset	Data	Source Line
03C9	0030	700 REM SWAP BACK TO THE MENU SCREEN (PAGE 0)
03CA	0030	SCREEN ,,0
03D6	0030	LOCATE 20,40
03E3	0030	
03E3	0030	705 GOSUB 2000 'REPLY FUNCTION
03E8	0030	
03E8	0030	710 REM DECODE THE ENTRY
03E9	0030	VALID\$="ABCDEFGHIX"
03F2	0034	BRANCH=INSTR (VALID\$,REPLY\$)
0400	003A	IF BRANCH = 0 THEN GOSUB 4000 : LOCATE 20,40 : GOTO 705
0421	003A	ON BRANCH GOSUB 800,810,820,830,835,840,845,850,860,870
043D	003A	GOTO 700
0441	003A	
0441	003A	800 REM FUNCTION 'A'.....OMR
0442	003A	CLOSE : CLS
044A	003A	CHAIN "RACP1A"
0451	003A	
0451	003A	810 REM FUNCTION 'B'.....
0452	003A	PRINT #1,"CLS"
045D	003A	PRINT #1,"ECHO ON"
0468	003A	PRINT #1,"REM PLACE IOMEGA CARTRIDGE IN DRIVE D"
0473	003A	PRINT #1,"REM "
047E	003A	PRINT #1,"REM VISIT.DAT FILE WILL BE COPIED TO DRIVE D"
0489	003A	PRINT #1,"REM AND RENAMED TO VISIT.OLD ON DRIVE C"
0494	003A	PRINT #1,"REM "
049F	003A	PRINT #1,"REM "
04AA	003A	PRINT #1,"PAUSE"
04B5	003A	PRINT #1,"ECHO OFF"
04C0	003A	PRINT #1,"COPY VISIT.DAT D:VISIT.DAT"
04CB	003A	PRINT #1,"IF EXIST VISIT.OLD ERASE VISIT.OLD"
04D6	003A	PRINT #1,"RENAME VISIT.DAT VISIT.OLD"
04E1	003A	PRINT #1,"CLS"
04EC	003A	PRINT #1,"RACB10"
04F7	003A	GOTO 9999
04FB	003A	RETURN
04FE	003A	
04FE	003A	820 REM FUNCTION 'C'
04FF	003A	PRINT #1,"CLS"
050A	003A	PRINT #1,"ECHO ON"
0515	003A	PRINT #1,"REM PLACE IOMEGA CARTRIDGE IN DRIVE D"
0520	003A	PRINT #1,"REM "
052B	003A	PRINT #1,"REM PATIENT.OUT FILE WILL BE COPIED TO DRIVE D"
0536	003A	PRINT #1,"REM AND RENAMED TO PATIENT.OLD ON DRIVE C"
0541	003A	PRINT #1,"REM "
054C	003A	PRINT #1,"REM "
0557	003A	PRINT #1,"PAUSE"
0562	003A	PRINT #1,"ECHO OFF"
056D	003A	PRINT #1,"COPY PATIENT.DAT D:PATIENT.OUT"
0578	003A	PRINT #1,"RENAME PATIENT.DAT PATIENT.OLD"
0583	003A	PRINT #1,"CLS"
058E	003A	PRINT #1,"RACB10"
0599	003A	GOTO 9999
059D	003A	
059D	003A	830 REM FUNCTION 'D'

RACP10
Initial Selection Screen

PAGE 7
J8-03-87
07:17:19

Offset Date Source Line

IBM Personal Computer BASIC Compiler V1.00

```

059E 003A      PRINT #1,"CLS"
05A9 003A      PRINT #1,"ECHO ON"
05B4 003A      PRINT #1,"REM      PLACE IOMEGA CARTRIDGE IN DRIVE D"
05BF 003A      PRINT #1,"REM "
05CA 003A      PRINT #1,"REM  PROVIDER.OUT FILE WILL BE COPIED TO DRIVE D"
05D5 003A      PRINT #1,"REM      AND RENAMED TO PROVIDER.OLD ON DRIVE C"
05E0 003A      PRINT #1,"REM "
05EB 003A      PRINT #1,"REM "
05F6 003A      PRINT #1,"PAUSE"
0601 003A      PRINT #1,"ECHO OFF"
060C 003A      PRINT #1,"COPY PROVIDER.DAT D:PROVIDER.OUT"
0617 003A      PRINT #1,"RENAME PROVIDER.DAT PROVIDER.OLD"
0622 003A      PRINT #1,"CLS"
062D 003A      PRINT #1,"RACP10"
0638 003A      GOTO 9999
063C 003A
063C 003A      835 REM FUNCTION 'E'
063D 003A      GOTO 700
0641 003A
0641 003A      840 REM FUNCTION 'F'
0642 003A      GOTO 700
0646 003A
0646 003A      845 REM FUNCTION 'G'
0647 003A      GOTO 700
064B 003A
064B 003A      850 REM FUNCTION 'H'
064C 003A      GOTO 700
0650 003A
0650 003A      860 REM FUNCTION 'I'
0651 003A      GOTO 700
0655 003A
0655 003A      870 REM FUNCTION 'X'      EXIT
0656 003A      GOTO 3010
065A 003A
065A 003A
065A 003A      2000 REM REPLY FUNCTION
065B 003A      2020 REPLY$=INKEY$ : IF REPLY$="" THEN 2020
066F 003A      REPLY=ASC(REPLY$)
0679 003C      IF REPLY > 90 THEN REPLY=CHR$(REPLY XOR 32)  'CONVERT TO CAPS
0694 003C      IF REPLY$ < "A" OR REPLY$ > "Z" THEN REPLY$="?"
06C0 003C      RETURN
06C3 003C
06C3 003C      3000 REM PF KEY TRAPS
06C4 003C
06C4 003C      3001 REM PF 1 = HELP
06C5 003C      SCREEN ,,1
06D2 003C      LOCATE 24,40
06DF 003C      GOSUB 2000      'REPLY FUNCTION
06E4 003C      RETURN 700
06EA 003C
06EA 003C      3002 REM PF 2 = UNUSED
06EB 003C      KEY (2) OFF      'DISABLE PF KEY
06F2 003C      RETURN 700
06F8 003C

```

Offset	Data	Source Line
06F8	003C	3003 REM PF 3 = UNUSED
06F9	003C	KEY (3) OFF 'DISABLE PF KEY
0700	003C	RETURN 700
0706	003C	
0706	003C	3004 REM PF 4 = UNUSED
0707	003C	KEY (4) OFF 'DISABLE PF KEY
070E	003C	RETURN 700
0714	003C	
0714	003C	3005 REM PF 5 = UNUSED
0715	003C	KEY (5) OFF 'DISABLE PF KEY
071C	003C	RETURN 700
0722	003C	
0722	003C	3006 REM PF 6 = UNUSED
0723	003C	KEY (6) OFF 'DISABLE PF KEY
072A	003C	RETURN 700
0730	003C	
0730	003C	3007 REM PF 7 = UNUSED
0731	003C	KEY (7) OFF 'DISABLE PF KEY
0738	003C	RETURN 700
073E	003C	
073E	003C	
073E	003C	3008 REM PF 8 = UNUSED
073F	003C	KEY (8) OFF 'DISABLE PF KEY
0746	003C	RETURN 700
074C	003C	
074C	003C	3009 REM PF 9 = UNUSED
074D	003C	KEY (9) OFF 'DISABLE PF KEY
0754	003C	RETURN 700
075A	003C	
075A	003C	3010 REM PF10 = LOGOFF, RESTART THIS PROGRAM
075B	003C	SCREEN ,,0
0767	003C	CLOSE
0768	003C	OPEN "O",1,"RACLOG.DAT"
0770	003C	CLOSE
0781	003C	OPEN "O",1,"RACBOO.BAT"
0793	003C	CLOSE
0797	003C	CHAIN "RACP05"
079E	003C	END
07A2	003C	
07A2	003C	4000 REM ERROR ON SELECTION
07A3	003C	LOCATE 21,1
07B0	003C	COLOR EFORE,EBACK
07BF	003C	FOR I = 1 TO 3
07C6	003C	SOUND 523.25,1.5
07D0	003C	SOUND 800.5,1.5
07DA	003C	NEXT
07E9	003C	PRINT " ERROR : ";REPLY\$;" INVALID OPTION " ;
07FB	003C	FOR I = 1 TO 3000
0802	003C	NEXT I
0812	003C	LOCATE 21,1
081F	003C	COLOR FORE,BACK
082E	003C	PRINT SPC(79)
0837	003C	RETURN
083A	003C	

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

```
083A 003C 7000 REM PRINT THE HEADING ON THE SCREEN
083B 003C REM $INCLUDE: 'RACS7000.SUB' INCLUDE THE STANDARD HEADER
083B 003C REM *****
083B 003C REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
083B 003C REM **** SKIP COLE ****
083B 003C REM **** SUBROUTINE NAME : RACS7000.SUB ****
083B 003C REM **** SCANNER PROGRAM # : ALL ****
083B 003C REM **** FUNCTION : THIS SUBROUTINE MODULE ****
083B 003C REM **** PRINTS THE STANDARD SCREEN ****
083B 003C REM **** HEADING. ****
083B 003C REM **** INPUT : COMMON VARIABLE USERS$(2) ****
083B 003C REM **** SYSTEM DATE ****
083B 003C REM ****
083B 003C REM **** OUTPUT : SCREEN HEADING ****
083B 003C REM ****
083B 003C REM **** RESERVED LINE ****
083B 003C REM **** NUMBERS : 7001-7010 ****
083B 003C REM *****
083B 003C
083B 003C 7001 LOCATE 1,1
0845 003C PRINT "U.S. ARMY AMBULATORY CARE INFORMATION SYSTEM"
084D 003C LOCATE 1,65
085A 003C PRINT DATES;
0862 003C LOCATE 2,1
086F 003C PRINT "USER : ";USERS$(1)
087C 003C RETURN
087F 003C
087F 003C 9999 REM EXIT TO THE BATCH FILE
0880 003C CLS
0884 003C SCREEN ,,0
0890 003C CLOSE
0894 003C END
0898 003C
089B 003C
```

22151 Bytes Available
18557 Bytes Free

0 Warning Error(s)
0 Severe Error(s)

PAGE 1

08-03-87

07:15:42

IBM Personal Computer BASIC Compiler V1.00

Offset Data Source Line

```
001A 0002 REM $LINESIZE: 132
001A 0002 REM $PAGESIZE: 66
001A 0002 REM $TITLE: 'RACP1A '
001A 0002 REM $SUBTITLE: 'OMR PROGRAM SELECT '
001A 0002 REM $PAGE
```

Offset Data Source Line

```

001A 0002  DEFINT A-Z
001A 0002  REM +-----+
001A 0002  REM | NAME: RACP1A          AMBULATORY CARE INFORMATION SYSTEM |
001A 0002  REM | DATE: 31 MAR 87          RE WRITTEN BY: D.R.BOLLING |
001A 0002  REM +-----+
001A 0002  REM          PATIENT OMR INPUT PROGRAM
001A 0002  REM
001A 0002  REM This program reads the OMR HEADER from the first record and
001A 0002  REM calls the proper program for decoding.....
001A 0002  REM
001A 0002  REM -----
001A 0002  REM $INCLUDE: 'RACDIM.MOD'      REM INCLUDE THE DIMENSION DEFINITIONS
001A 0002  REM *****
001A 0002  ' *   NAME: RACDIM.MOD          DIMENSION DEFINITIONS   *
001A 0002  ' *   Date: 28 Feb 84          Written by: Floyd Cole   *
001A 0002  REM *****
001A 0002  '   Dimensioned variables are defined in this file.
001A 0002  '   It is an included file so it cannot be run in a stand-alone,
001A 0002  '   mode.
001A 0002  '
001A 0002  '   This program segment may be modified, but all files containing
001A 0002  '   an include for this segment must be re-compiled in order to
001A 0002  '   affect the changes made here.
001A 0002  '   ***** START OF DIMENSION DEFINITION *****
001A 0002
001A 0002  DEFINT A-Z
001A 0002  DIM USERS$(2),MOLENGTH(12),DATEERR$(3)
001A 0002
001A 0002  '   ***** END OF DIMENSION DEFINITIONS *****
001A 0002
001A 0002  DIM SHEETREC(1750)  '(MAX. SIZE FOR A SHEET FROM THE SCANNER)
001A 0002  DIM PROTOCOL(7)      '(ARRAY FOR SERIAL BOARD SETUP PARAMETERS)
001A 0002
001A 0002  REM $INCLUDE: 'RACCMN.MOD'      REM INCLUDE THE COMMON AREA DEFINITION
001A 0002  REM *****
001A 0002  ' *   NAME: RACCMN.MOD          COMMON AREA DEFINITION   *
001A 0002  ' *   Date: 28 Feb 84          Written by: Floyd Cole   *
001A 0002  REM *****
001A 0002  '   COMMON AREA DEFINITIONS WILL BE HELD IN THIS FILE. IT IS AN
001A 0002  '   INCLUDED FILE SO IT CANNOT BE RUN IN A STAND*ALONE, MODE.
001A 0002  '
001A 0002  '   This program segment may be modified, but all files containing
001A 0002  '   an include for this segment must be re*compiled in order to
001A 0002  '   affect the changes made here.
001A 0002  '
001A 0002  '   *****START OF COMMON DEFINITIONS*****
001A 0002
001A 0002  COMMON FORE,BACK,BOARD,HIDE,EFORE,EBACK,BELL$ 'BASIC SCREEN COLORS
001A 0002  COMMON HEADERS          '21 CHARACTER SCANNER HEADER INFO
001A 0002  COMMON TEXT$           '11 AINING CHARACTERS FROM SCANNER
001A 0002  COMMON PGMIDS        'PROGRAM OR FORM ID
001A 0002  COMMON MOLENGTH( )  'DAYS IN THE MONTH
001A 0002  COMMON USERS( )

```

```

Offset Data Source Line
001A 0002 ' *****END OF COMMON DEFINITION*****
001A 0002
061A 0002
001A 0002 REM $INCLUDE: 'RACDEF.MOD' REM INCLUDE THE DEFAULT DEFINITIONS
001A 0002 '*****
001A 0002 '* NAME: RACP01.DEF DEFAULT DEFINITIONS *
001A 0002 '* Date: 28 Feb 84 Written by: Floyd Cole *
001A 0002 '*****
001A 0002 ' Variables used in common that have a default value on start*up
001A 0002 ' will be held in this file. It is an included file so it cannot
001A 0002 ' be run in a stand*alone mode. In normal operation, this file
001A 0002 ' should be 'included' in the main program only (RACP10.BAS).
001A 0002 '
001A 0002 ' This program segment may be modified, but all files containing
001A 0002 ' an include for this segment must be re*compiled in order to
001A 0002 ' affect the changes made here.
001A 0002 '
001A 0002 ' *****START OF DEFAULT DEFINITION*****
001A 0002 FORE = 15 'FOREGROUND COLOR = INTENSE WHITE
0040 0002 BACK = 1 'Background Color = Light Blue
0047 0002 BORD = 4 'BORDER = RED
004E 0002 HIDE = 4 'ALTERNATE COLOR = RED
0055 0002 EFORE = 14 'ERROR FOREGROUND DISPLAY
005C 0002 EBACK = 0 'ERROR BACKGROUND DISPLAY
0063 0002 BELLS = CHR$(7) 'Sound the bell
006F 0002
006F 0002 MOLENGTH(1) = 31 'JAN
0076 0002 MOLENGTH(2) = 28 'FEB <--MODIFIED IN SUBROUTINE RACS5000.SUB
007D 0002 MOLENGTH(3) = 31 'MAR
0084 0002 MOLENGTH(4) = 30 'APR
0088 0002 MOLENGTH(5) = 31 'MAY
0092 0002 MOLENGTH(6) = 30 'JUN
0099 0002 MOLENGTH(7) = 31 'JUL
00A0 0002 MOLENGTH(8) = 31 'AUG
00A7 0002 MOLENGTH(9) = 30 'SEP
00AE 0002 MOLENGTH(10) = 31 'OCT
00B5 0002 MOLENGTH(11) = 30 'NOV
00BC 0002 MOLENGTH(12) = 31 'DEC
00C3 0002
00C3 0002 DATEERR$(0) = " "
00CC 0002 DATEERR$(1) = "INVALID MONTH"
00D5 0002 DATEERR$(2) = "INVALID DAY "
00DE 0002 DATEERR$(3) = "DAY TOO LARGE FOR MONTH CODED"
00E7 0002
00E7 0002 MAXLENGTH = 80 'MAXIMUM LENGTH OF OUTPUT RECORD
00EE 0002 PAD$ = ". " 'PAD CHARACTER FOR SHORT RECORDS
00F7 0008
00F7 0008 ' *****END OF DEFAULT DEFINITION*****
00F7 0008
00F7 0008 KEY OFF
00FD 0008 REM LENGTH OF STRING RECEIVED FROM THE OMR....

```

Offset	Data	Source Line
00FD	00D8	HEADER = 21
0104	00DA	RECORDLENGTH = HEADER
0108	00DC	
0108	00DC	REM -----
0108	00DC	REM PAGE 1 - THE SCREEN HEADER
0108	00DC	REM -----
0108	00DC	20 CLS
010F	00DC	COLOR 14
0116	00DC	LOCATE 12,25 : PRINT " START THE SCANNER "
0128	00DC	LOCATE 15,25 : PRINT " PRESS ANY KEY TO BEGIN "
0140	00DC	
0140	00DC	GOSUB 2000 'DELAY
0145	00DC	
0145	00DC	REM *****
0145	00DC	REM **** COMMUNICATIONS SETUP ****
0145	00DC	REM *****
0145	00DC	REM
0145	00DC	GOSUB 9001
014A	00DC	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
0164	0DE0	
0164	0DE0	REM *****
0164	0DE0	REM **** START THE SCANNER ****
0164	0DE0	REM *****
0164	0DE0	
0164	0DE0	CNTRLOPT =1 :GOSUB 9010 'SI - START THE READER
0170	0DE2	IF ERRMSG\$ > " " THEN LPRINT ERRMSG\$: GOTO 30000
018A	0DE2	
018A	0DE2	REM *****
018A	0DE2	REM **** SET SCAN SHEET CALL ****
018A	0DE2	REM *****
018A	0DE2	REM
018A	0DE2	
018A	0DE2	200 REM - FIRST SHEET TRANSMIT
0188	0DE2	
0188	0DE2	READTYPE = 2
0192	0DE4	GOSUB 9020 'SCAN SUBROUTINE
0197	0DE4	IF MID\$(ERRSTAT\$,14,3) = "415" GOTO 25000
01B3	0DE8	
01B3	0DE8	REM *****
01B3	0DE8	REM **** DECODE THE HEADER ****
01B3	0DE8	REM *****
01B3	0DE8	210 REM
01B4	0DE8	TEXT\$="" 'CLEAR THE INPUT AREA
01B0	0DE8	RECORDPTR = VARPTR(SHEETREC(0))
01C4	0DEA	TOT = 0
01C8	0DEC	HEADERS\$=""
01D4	0DEC	FOR JA1 = 1 TO RECORDLENGTH
01E1	0DEE	TEXT\$= TEXT\$+CHR\$(PEEK(RECORDPTR + TOT))
01FF	0DEE	TOT=TOT+1
0207	0DEE	NEXT JA1
0218	0DF0	
0218	0DF0	PROGRAM\$= LEFT\$(TEXT\$,3)
0227	0DF4	PGMID\$= PROGRAM\$
0230	0DF4	BATCH\$= MID\$(TEXT\$,4,3)

Offset	Data	Source Line
0242	0DF8	SERIAL\$= MID\$(TEXT\$,7,4)
0254	0DFC	RUNID\$= MID\$(TEXT\$,11,1)
0266	0E00	FORM\$= MID\$(TEXT\$,12,2)
0278	0E04	POCKET\$= MID\$(TEXT\$,14,1)
028A	0E08	SCANERR1\$=MID\$(TEXT\$,16,2)
029C	0E0C	SCANERR2\$=MID\$(TEXT\$,18,2)
02AE	0E10	SCANERR3\$=MID\$(TEXT\$,20,2)
02C0	0E14	
02C0	0E14	REM *****
02C0	0E14	REM **** WHAT PROGRAM IS IT?? ****
02C0	0E14	REM *****
02C0	0E14	
02C0	0E14	IF PROGRAM\$ = "200" GOTO 320 'PATIENT REGISTRATION
02D2	0E14	IF PROGRAM\$ = "210" GOTO 325 'QUALITY CONTROL PROG
02E4	0E14	IF PROGRAM\$ = "300" GOTO 330 'PROVIDER REGISTRATION
02F6	0E14	IF PROGRAM\$ = "400" GOTO 400 'ENCOUNTER FORM .. ADOLESCENT MED
0308	0E14	IF PROGRAM\$ = "410" GOTO 400 'ENCOUNTER FORM .. ALLERGY
031A	0E14	IF PROGRAM\$ = "420" GOTO 400 'ENCOUNTER FORM .. AUDIOLOGY/SPEE
032C	0E14	IF PROGRAM\$ = "430" GOTO 400 'ENCOUNTER FORM .. CARDIOLOGY
033E	0E14	IF PROGRAM\$ = "440" GOTO 400 'ENCOUNTER FORM .. CARDIOTHORACIC
0350	0E14	IF PROGRAM\$ = "450" GOTO 400 'ENCOUNTER FORM .. DERMATOLOGY
0362	0E14	IF PROGRAM\$ = "460" GOTO 400 'ENCOUNTER FORM .. ENDOCRINE
0374	0E14	IF PROGRAM\$ = "470" GOTO 400 'ENCOUNTER FORM .. ENT
0386	0E14	IF PROGRAM\$ = "480" GOTO 400 'ENCOUNTER FORM .. FAMILY MED
0398	0E14	IF PROGRAM\$ = "490" GOTO 400 'ENCOUNTER FORM .. GASTROENTEROLO
03AA	0E14	IF PROGRAM\$ = "500" GOTO 400 'ENCOUNTER FORM .. GENERAL SURG
03BC	0E14	IF PROGRAM\$ = "510" GOTO 400 'ENCOUNTER FORM .. GYNECOLOGY
03CE	0E14	IF PROGRAM\$ = "520" GOTO 400 'ENCOUNTER FORM .. INFECTIOUS DIS
03E0	0E14	IF PROGRAM\$ = "530" GOTO 400 'ENCOUNTER FORM .. INTERNAL MED
03F2	0E14	IF PROGRAM\$ = "540" GOTO 400 'ENCOUNTER FORM .. NEPHROLOGY
0404	0E14	IF PROGRAM\$ = "550" GOTO 400 'ENCOUNTER FORM .. NEUROLOGY
0416	0E14	IF PROGRAM\$ = "560" GOTO 400 'ENCOUNTER FORM .. NEUROSURGERY
0428	0E14	IF PROGRAM\$ = "570" GOTO 400 'ENCOUNTER FORM .. NUCLEAR MED
043A	0E14	IF PROGRAM\$ = "580" GOTO 400 'ENCOUNTER FORM .. NUTRITION
044C	0E14	IF PROGRAM\$ = "590" GOTO 400 'ENCOUNTER FORM .. OBSTETRICS
045E	0E14	IF PROGRAM\$ = "600" GOTO 400 'ENCOUNTER FORM .. ONCOLOGY/HEM
0470	0E14	IF PROGRAM\$ = "610" GOTO 400 'ENCOUNTER FORM .. OPHTHAMOLOGY
0482	0E14	IF PROGRAM\$ = "620" GOTO 400 'ENCOUNTER FORM .. OPTOMETRY
0494	0E14	IF PROGRAM\$ = "630" GOTO 400 'ENCOUNTER FORM .. ORTHO APPL
04A6	0E14	IF PROGRAM\$ = "640" GOTO 400 'ENCOUNTER FORM .. ORTHOPEDICS
04B8	0E14	IF PROGRAM\$ = "650" GOTO 400 'ENCOUNTER FORM .. OCCUPAT THER
04CA	0E14	IF PROGRAM\$ = "660" GOTO 400 'ENCOUNTER FORM .. PAIN/PHYS MED
04DC	0E14	IF PROGRAM\$ = "670" GOTO 400 'ENCOUNTER FORM .. PEDIATRICS
04EE	0E14	IF PROGRAM\$ = "680" GOTO 400 'ENCOUNTER FORM .. PHYS THERAPY
0500	0E14	IF PROGRAM\$ = "690" GOTO 400 'ENCOUNTER FORM .. PLASTIC SUR
0512	0E14	IF PROGRAM\$ = "700" GOTO 400 'ENCOUNTER FORM .. PODIATRY
0524	0E14	IF PROGRAM\$ = "710" GOTO 400 'ENCOUNTER FORM .. PREV MED
0536	0E14	IF PROGRAM\$ = "720" GOTO 400 'ENCOUNTER FORM .. PRIMARY CARE
0548	0E14	IF PROGRAM\$ = "730" GOTO 400 'ENCOUNTER FORM .. PSYCHIATRY
055A	0E14	IF PROGRAM\$ = "740" GOTO 400 'ENCOUNTER FORM .. PSYCHOLOGY
056C	0E14	IF PROGRAM\$ = "750" GOTO 400 'ENCOUNTER FORM .. PULMONARY
057E	0E14	IF PROGRAM\$ = "760" GOTO 400 'ENCOUNTER FORM .. RADIOTHERAPY
0590	0E14	IF PROGRAM\$ = "770" GOTO 400 'ENCOUNTER FORM .. RHEUMATOLOGY
05A2	0E14	IF PROGRAM\$ = "780" GOTO 400 'ENCOUNTER FORM .. SOCIAL WORK

Offset	Data	Source Line
05B4	0E14	IF PROGRAMS = "790" GOTO 400 'ENCOUNTER FORM .. UROLOGY
05C6	0E14	IF PROGRAMS = "800" GOTO 800 'ENCOUNTER FORM .. BAS/TMC
05D8	0E14	IF PROGRAMS = "810" GOTO 810 'ENCOUNTER FORM .. EFMP
05EA	0E14	IF PROGRAMS = "820" GOTO 820 'ENCOUNTER FORM .. EKG
05FC	0E14	IF PROGRAMS = "830" GOTO 830 'ENCOUNTER FORM .. EMERGENCY RM
060E	0E14	IF PROGRAMS = "840" GOTO 840 'ENCOUNTER FORM .. GROUP I
0620	0E14	IF PROGRAMS = "850" GOTO 850 'ENCOUNTER FORM .. GROUP II
0632	0E14	IF PROGRAMS = "860" GOTO 860 'ENCOUNTER FORM .. IMMUNIZ SHORT
0644	0E14	IF PROGRAMS = "870" GOTO 870 'ENCOUNTER FORM .. IMR
0656	0E14	IF PROGRAMS = "880" GOTO 880 'ENCOUNTER FORM .. INHALATION TH
0668	0E14	IF PROGRAMS = "890" GOTO 890 'ENCOUNTER FORM .. NUTRITION SHOR
067A	0E14	IF PROGRAMS = "900" GOTO 900 'ENCOUNTER FORM .. OCCUP HEALTH
068C	0E14	IF PROGRAMS = "910" GOTO 910 'ENCOUNTER FORM .. OCCUP THER RPT
069E	0E14	IF PROGRAMS = "920" GOTO 920 'ENCOUNTER FORM .. PHYS TH REPEAT
06B0	0E14	IF PROGRAMS = "930" GOTO 930 'ENCOUNTER FORM .. REPEAT
06C2	0E14	IF PROGRAMS = "940" GOTO 940 'ENCOUNTER FORM .. SHORT
06D4	0E14	IF PROGRAMS = "950" GOTO 950 'ENCOUNTER FORM .. SOC WORK SHORT
06E6	0E14	
06E6	0E14	CLS : LOCATE 12,10 : BEEP
06FB	0E14	PRINT "PROGRAM ";PROGRAMS;" IS NOT A VALID CODE.."
070D	0E14	PRINT " MAKE SURE YOU HAVE LOADED THE FORMS CORRECTLY.."
0715	0E14	PRINT " PRESS ANY KEY TO RETURN TO THE MAIN MENU"
071D	0E14	GOSUB 2000
0722	0E14	GOTO 30000
0726	0E14	
0726	0E14	
0726	0E14	320 CLS : CLOSE
072E	0E14	CHAIN "RACP200" 'PATIENT REGISTRATION
0735	0E14	
0735	0E14	325 CLS : CLOSE
073D	0E14	CHAIN "RACPVER" 'QC VERIFICATION PROGRAM
0744	0E14	
0744	0E14	330 CLS : CLOSE
074C	0E14	CHAIN "RACP300" 'PROVIDER REGISTRATION
0753	0E14	
0753	0E14	400 CLS : CLOSE
075B	0E14	CHAIN "RACPGEN" 'GENERAL FORM PROGRAM
0762	0E14	
0762	0E14	800 CLS : CLOSE
076A	0E14	CHAIN "RACP800" 'BAS/TMC FORM PROGRAM
0771	0E14	
0771	0E14	810 CLS : CLOSE
0779	0E14	CHAIN "RACP810" 'EFMP FORM PROGRAM
0780	0E14	
0780	0E14	820 CLS : CLOSE
0788	0E14	CHAIN "RACP820" 'EKG FORM PROGRAM
078F	0E14	
078F	0E14	830 CLS : CLOSE
0797	0E14	CHAIN "RACP830" 'EMERGENCY ROOM PROGRAM
079E	0E14	
079E	0E14	840 CLS : CLOSE
07A6	0E14	CHAIN "RACP840" 'GROUP I FORM PROGRAM
07AD	0E14	
07AD	0E14	850 CLS : CLOSE

Offset	Data	Source Line
07B5	0E14	CHAIN "RACP850" 'GROUP II FORM PROGRAM
07BC	0E14	
078C	0E14	860 CLS : CLOSE
07C4	0E14	CHAIN "RACP860" 'IMMUNIZATION SHORT FORM PROG
07CB	0E14	
07CB	0E14	870 CLS : CLOSE
07D3	0E14	CHAIN "RACP870" 'IMR FORM PROGRAM
07DA	0E14	
07DA	0E14	880 CLS : CLOSE
07E2	0E14	CHAIN "RACP880" 'INHALATION THERAPY FORM PROG
07E9	0E14	
07E9	0E14	890 CLS : CLOSE
07F1	0E14	CHAIN "RACP890" 'NUTRITION CARE FORM PROGRAM
07F8	0E14	
07F8	0E14	900 CLS : CLOSE
0800	0E14	CHAIN "RACP900" 'OCCUPATIONAL HEALTH FORM PROGRAM
0807	0E14	
0807	0E14	910 CLS : CLOSE
080F	0E14	CHAIN "RACP910" 'OCCUPATIONAL THERAPY PROGRAM
0816	0E14	
0816	0E14	920 CLS : CLOSE
081E	0E14	CHAIN "RACP920" 'PHYS THERAPY FORM PROGRAM
0825	0E14	
0825	0E14	930 CLS : CLOSE
082D	0E14	CHAIN "RACP930" 'REPEAT FORM PROGRAM
0834	0E14	
0834	0E14	940 CLS : CLOSE
083C	0E14	CHAIN "RACP940" 'SHORT FORM PROGRAM
0843	0E14	
0843	0E14	950 CLS : CLOSE
0848	0E14	CHAIN "RACP950" 'SOCIAL WORK SHORT FORM PROGRAM
0852	0E14	
0852	0E14	REM END OF SCAN/DECODE/WRITE LOOP =====
0852	0E14	
0852	0E14	
0852	0E14	2000 REM \$INCLUDE: 'RACS2000.SUB' INCLUDE THE DELAY SUBROUTINE
0853	0E14	REM *****
0853	0E14	REM **** AMBULATORY CARE DATA BASE 13 APR 85 ****
0853	0E14	REM **** SKIP COLE ****
0853	0E14	REM **** SUBROUTINE NAME : RACS2000.SUB ****
0853	0E14	REM **** SCANNER PROGRAM # : ALL ****
0853	0E14	REM **** FUNCTION : THIS SUBROUTINE MODULE ****
0853	0E14	REM **** SERVERS AS A WAIT AND REPLY ****
0853	0E14	REM **** ENTRY MODULE ****
0853	0E14	REM **** INPUT : SINGLE KEYBOARD ENTRY ****
0853	0E14	REM **** OUTPUT : KEYBOARD ENTRY - UPPER CASE ****
0853	0E14	REM **** RESERVED LINE ****
0853	0E14	REM **** NUMBERS : 2001-2010 ****
0853	0E14	REM *****
0853	0E14	2001 REM REPLY FUNCTION
0854	0E14	2002 REPLY\$=INKEY\$: IF REPLY\$="" THEN 2002
0868	0E18	REPLY=ASC(REPLY\$)

Offset	Date	Source Line
0872	0E1A	IF REPLY > 90 THEN REPLY\$=CHR\$(REPLY XOR 32) 'CONVERT TO CAPS
0880	0E1A	IF REPLY\$ < "A" OR REPLY\$ > "Z" THEN REPLY\$="?"
0889	0E1A	RETURN
088C	0E1A	
088C	0E1A	9000 REM \$INCLUDE: 'RACS9000.SUB' INCLUDE THE SCANNER CONTROL SUBS
088D	0E1A	REM *****
088D	0E1A	**** AMBULATORY CARE DATA BASE 13 APR 85 ****
088D	0E1A	**** SKIP COLE ****
088D	0E1A	**** PROGRAM NAME : RACS9000.SUB ****
088D	0E1A	**** SCANNER PROGRAM # : ALL ****
088D	0E1A	**** FUNCTION : THIS SUBROUTINE MODULE ****
088D	0E1A	**** CONTROLS THE SCANNER I/O ****
088D	0E1A	****
088D	0E1A	**** INPUT/OUTPUT : REFER TO THE ASYNCHRONOUS ****
088D	0E1A	**** COMMUNICATIONS MANUAL AND THE ****
088D	0E1A	**** PRE-RELEASED SOFTWARE GUIDE ****
088D	0E1A	****
088D	0E1A	*****
088D	0E1A	**** RESERVED LINE ****
088D	0E1A	**** NUMBERS : 9001-9100 ****
088D	0E1A	*****
088D	0E1A	
088D	0E1A	*****
088D	0E1A	**** SUBROUTINE 9001 - PROTOCOL SETUP FOR SCANNER ****
088D	0E1A	**** ARGUMENTS: PRESET ... SEE BELOW ****
088D	0E1A	*****
088D	0E1A	9001 REM
088E	0E1A	PROTOCOL(0) = 9600 'BAUD RATE
08C5	0E1A	PROTOCOL(1) = 78 'PARITY (SEE PAGE 4-8 OF MANUAL)
08CC	0E1A	PROTOCOL(2) = 8 'DATA BITS
08D3	0E1A	PROTOCOL(3) = 1 'STOP BITS
08DA	0E1A	PROTOCOL(4) = 2 'RS-232 PORT
08E1	0E1A	PROTOCOL(5) = 0 'WRITE TIME-OUT
08E8	0E1A	PROTOCOL(6) = 0 'READ TIME-OUT
08EF	0E1A	
08EF	0E1A	ERRSTAT\$ = SPACE\$(60)
08FB	0E1A	ARGPTR = VARPTR(PROTOCOL(0))
0902	0E1C	CALL SETUP (ARGPTR,ERRSTAT\$)
0913	0E1C	ERRMSG\$=""
091C	0E1C	IF ASC(ERRSTAT\$) <> 64 THEN ERRMSG\$="SETUP ERROR "+ERRSTAT\$
0938	0E1C	GOTO 9100
093C	0E1C	
093C	0E1C	*****
093C	0E1C	**** SUBROUTINE 9010 - CONTROL OPTIONS FOR SCANNER ****
093C	0E1C	**** ARGUMENTS: CNTRLOPT ****
093C	0E1C	**** CNTRLOPT = 1 = START SCANNER (S1) ****
093C	0E1C	**** CNTRLOPT = 2 = STOP SCANNER (S0) ****
093C	0E1C	**** CNTRLOPT = 3 = TERMINATE COMMUNICATIONS TO SCANNER (DC3) ****
093C	0E1C	**** CNTRLOPT = 4 = CLEAR TRANSPORT PATH (DC2) ****
093C	0E1C	**** CNTRLOPT = 5 = SELECT PRIMARY STACKER "31" ****
093C	0E1C	**** CNTRLOPT = 6 = SELECT SECONDARY STACKER "32" ****
093C	0E1C	**** CNTRLOPT = 7 = POSITIVE RESPONSE/SELECT SCANNER (DC1) ****
093C	0E1C	**** CNTRLOPT = 8 = REQUEST STATUS (ESC) ****

Offset Data Source Line

```

093C 0E1C REM *****
093C 0E1C 9010 REM
093D 0E1C ERRSTAT$ = SPACE$(60)
0949 0E1C CALL CNTRL (CNTRLOPT,ERRSTAT$)
095A 0E1C ERRMSG$=""
0963 0E1C IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="CONTROL ERROR "+ERRSTAT$
097F 0E1C GOTO 9100
0983 0E1C REM *****
0983 0E1C REM **** SUBROUTINE 9020 - SCAN SHEET CALL ****
0983 0E1C REM ****
0983 0E1C REM **** ARGUMENTS: READTYPE ****
0983 0E1C REM **** READTYPE = 2 = REQUEST NEW DOCUMENT FROM SCANNER ****
0983 0E1C REM **** READTYPE = 3 = RETRANSMIT CURRENT DOCUMENT ****
0983 0E1C REM ****
0983 0E1C REM **** ARGUMENTS: RECORDLENGTH ****
0983 0E1C REM **** NUMERIC VARIABLE SET TO THE NUMBER OF CHARACTERS TO BE ****
0983 0E1C REM **** TRANSMITTED ****
0983 0E1C REM *****
0983 0E1C 9020 REM
0984 0E1C ERRSTAT$ = SPACE$(60)
0990 0E1C RECORDPTR = VARPTR(SHEETREC(0))
0997 0E1C CALL SCAN (READTYPE,RECORDLENGTH,RECORDPTR,ERRSTAT$)
0980 0E1C ERRMSG$=""
0989 0E1C IF MID$(ERRSTAT$,14,3) = "415" THEN ERRMSG$="ESC"
09DA 0E1C GOTO 9100
09DE 0E1C REM *****
09DE 0E1C REM **** SUBROUTINE 9030 - TRANSPORT PRINT CALL ****
09DE 0E1C REM ****
09DE 0E1C REM **** ARGUMENTS: PRINTPOS ****
09DE 0E1C REM **** NUMERIC VARIABLE INDICATING THE STARTING PRINT POSITION ****
09DE 0E1C REM **** VALUES = 0 THRU 90 ****
09DE 0E1C REM ****
09DE 0E1C REM **** ARGUMENTS: PSTRINGS ****
09DE 0E1C REM **** TEXT TO BE PRINTED ON THE FORM ****
09DE 0E1C REM ****
09DE 0E1C REM **** NOTE: THIS ROUTINE HAS NO EFFECT UNLESS THE SCAN ****
09DE 0E1C REM **** HEADER SHEET IS MARKED 'PRINTER ON' ****
09DE 0E1C REM *****
09DE 0E1C 9030 REM
09DF 0E1C ERRSTAT$ = SPACE$(60)
09E8 0E1C RECORDPTR = VARPTR(SHEETREC(0))
09F2 0E1C CALL TPRINT(PRINTPOS,PSTRINGS$,ERRSTAT$)
0A07 0E22 IF ASC(ERRSTAT$) <> 64 THEN ERRMSG$="PRINT ERROR "+ERRSTAT$
0A23 0E22 GOTO 9100
0A27 0E22
0A27 0E22 9100 RETURN
0A2A 0E22 REM -----END OF SUBROUTINE RACS9000.SUB -----
0A2A 0E22
0A2A 0E22 25000 REM USER TERMINATED INPUT
0A2B 0E22 CLS : BEEP
0A33 0E22 LOCATE 14,28:PRINT "PROGRAM TERMINATED BY USER"
0A48 0E22

```

RACP1A
OMR PROGRAM SELECT

PAGE 10
08-03-87
07:15:42

Offset Data Source Line

IBM Personal Computer BASIC Compiler V1.00

0A48	0E22	30000 REM
0A49	0E22	CLOSE
0A4D	0E22	CHAIN "RACP10"
0A54	0E22	
0A57	0E22	

22151 Bytes Available
18526 Bytes Free

0 Warning Error(s)
0 Severe Error(s)

APPENDIX D

PC FOCUS PROCEDURES (REFERENCED SECTION 4.5)

APPENDIX D

PC FOCUS PROCEDURES

TABLE OF CONTENTS

CLINIC.MAS	Clinic Master file PC
CLINIC1.FEX	Report 1 Provider by Clinic
CLINIC2.FEX	Report 2 Patient Cat by Clinic
CLINIC3.FEX	Report 3 Referral Source
CLINIC4.FEX	Report 9 Job related
DEERS.FEX	DEERS Special Load PC
DIAGDMP.FEX	Diagnosis Description Dump PC
DIAGNOS.MAS	Diagnosis Master file PC
LOADCLIN.FEX	Clinic Description Load PC
LOADDIAG.FEX	Diagnosis Description Load PC
LOADOTH.FEX	Other Code Description Load PC
LOADPROC.FEX	Procedure Description Load PC
MAIN.FEX	Main menu program PC
MILT1.FEX	Report 10 Military Unit
MILT2.FEX	Report 11 Unit Health
MILT3.FEX	Report 12 (unfinished)
OTHER.MAS	Other Codes Master file PC
PATIENT.MAS	Patient Master file PC
PROCDMP.FEX	Procedure Description Dump PC
PROCEDUR.MAS	Procedure Master file PC
PROFILE.FEX	FOCUS Initialization file PC
PROV1.FEX	Report 4 Provider Diagnosis
PROV2.FEX	Report 5 Provider Procedure
PROV3.FEX	Report 6 Provider Time/Status
PROV4.FEX	Report 7 Provider Dx by Clinic
PROVIDER.MAS	Provider Master file PC
RPTCRT.FEX	Selection Menu for Rpts 1-6
RPTCRT1.FEX	Selection Menu for Rpts 7,8
RPTCRT2.FEX	Selection Menu for Rpts 10
RPTCRT3.FEX	Selection Menu for Rpts 11
RPTCRT4.FEX	Selection Menu for Rpts 12
RPTCRT5.FEX	Selection Menu for Rpts 9
RXXCCLD.FEX	Clinic Correction Dump PC
RXXCORL.FEX	Loads Corrected file to Visit
RXXCPAD.FEX	Patient Correction Dump PC
RXXCPRD.FEX	Provider Correction Dump PC
RXXF200.FEX	Patient Load file PC
RXXF300.FEX	Provider Load file PC
RXXF400.FEX	Visit Load file PC
RXXFCCL.FEX	Control Clinic Correction PC
RXXFCPA.FEX	Control Patient Correction PC
RXXFCPR.FEX	Control Provider Correction PC
RXXFDCL.FEX	Clinic Dump file PC
RXXFDPA.FEX	Patient Dump file PC
RXXFDPR.FEX	Provider Dump file PC

RXXFDVI.FEX Visit Dump file PC
VISIT.MAS Visit Master file PC

CLINIC.MAS

FILENAME=CLINIC,SUFFIX=FOC

SEGNAME=CLINIC,SEGTYPE=S1

FIELDNAME=CLINIC,	ALIAS=CLUCA,	FORMAT=A4,	FIELDTYPE=I,\$
FIELDNAME=CL_TITLE,	ALIAS=302CLTITLE,	FORMAT=A20	,\$
FIELDNAME=CL_302CODE,	ALIAS=302CLCODE,	FORMAT=A2	,\$
FIELDNAME=CL_302LINE,	ALIAS=302CLLINE,	FORMAT=I3	,\$

END

DBA=LUGNUT,\$

USER=NUTMEG, ACCESS=RW, \$

```

-*****
-*
-* PROGRAM: CLINIC1.FEX      MODIFIED: 6/17/87 DRB  *
-*                                FOR NEW DESIGN    *
-* REPORT: 1                  MENU PROGRAM: RPTCRT.FEX *
-*
-*****
-*
-SET &&RPT='ENCOUNTERS PER PROVIDER PER CLINIC COUNT';
EX RPTCRT
-RUN
-IF &&RTN EQ 'X' THEN GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&CLN=&CLN;
-SET &&PROV=&PROV;
-CONT1
JOIN PROV_ID IN VISIT TO PROV_ID IN PROVIDER AS JOIN1
JOIN CLINIC IN VISIT TO CLINIC IN CLINIC AS JOIN2
TABLE FILE VISIT
  HEADING CENTER
  " ENCOUNTERS PER PROVIDER PER CLINIC COUNT"
  " FOR &&ST TO &&END </2"
SUM CNT LITHO AS 'VISIT,COUNT' IN 60
  VISIT_CNT/I6 AS 'WEIGHTED,COUNT' IN 70
BY CLINIC AS 'CLINIC, CODE'
BY CL_TITLE AS 'CLINIC TITLE' IN 10
BY PROV_ID AS 'PROVIDER, ID.' IN 35
BY PROV_LST_NME AS 'LAST NAME' IN 45
IF CLINIC EQ &&CLN
IF VISIT_DATE FROM &&ST TO &&END
IF PROV_ID EQ &&PROV
ON CLINIC SUBTOTAL AS 'TOTAL VISITS FOR CLINIC'
ON CLINIC PAGE-BREAK
END
JOIN CLEAR *
-RUN

```

```

*****
-*
-* PROGRAM: CLINIC2.FEX          MODIFIED: 6/17/87 DRB
-*                                FOR NEW DESIGN
-* REPORT: 2                     MENU PROGRAM: RPTCRT.FEX
-*
*****
-*
-SET &&RPT = 'PATIENT CATEGORY BY CLINIC';
EX RPTCRT
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&CLN=&CLN;
-SET &&PROV=&PROV;
-CONT1
JOIN CLINIC IN VISIT TO CLINIC IN CLINIC AS JJOIN
JOIN PTID      IN VISIT TO PTID      IN PATIENT AS JJOIN1
DEFINE FILE VISIT
WN/A7=IF PT_CATEGORY EQ ' ' THEN 'N O N E' ELSE PT_CATEGORY;
END
TABLE FILE VISIT
HEADING CENTER
"P A T I E N T   C A T E G O R Y   B Y   C L I N I C"
"FOR <CL_TITLE"
"FROM &&ST TO &&END </2"
COUNT LITHO_CODE          AS 'PATIENT,COUNT' IN 85
BY CLINIC IN 35 BY CL_TITLE AS ' CLINIC, NAME ' IN 45
BY WN AS 'PATIENT,CATEGORY' IN 70
IF CLINIC EQ &&CLN
IF VISIT_DATE FROM &&ST TO &&END
IF FNUM NE '84'
ON CLINIC SUBTOTAL PAGE-BREAK
FOOTING CENTER
"N O N E INDICATES THOSE PATIENTS ARE NOT REGISTERED."
END
JOIN CLEAR *
-RUN

```

```

-*****
-*
-* PROGRAM: CLINIC3.FEX          MODIFIED: 6/17/87  DRB  *
-*                               FOR NEW DESIGN          *
-* REPORT: 3                     MENU PROGRAM: RPTCRT.FEX *
-*
-*****
-*
-SET &&RPT='REFERRAL SOURCE AND PLACE OF VISIT';
-RUN
EX RPTCRT
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&CLN=&CLN;
-SET &&PROV=&PROV;
-CONT1
JOIN CLINIC IN VISIT TO CLINIC IN CLINIC AS JOIN1
-INCLUDE DCODE2
-RUN
DEFINE FILE VISIT ADD
CLTITLE/A20=IF CL_TITLE EQ ' ' THEN CLINIC ELSE CL_TITLE;
END
TABLE FILE VISIT
SUM CNT LITHO COLUMN-TOTAL AS '  NUMBER,          OF,ENCOUNTERS' IN 25

      VISIT_CNT/I6 COLUMN-TOTAL AS 'WEIGHTED,NO. OF,ENCOUNTERS' IN 36
BY CLINIC  NOPRINT
BY CLTITLE AS '      CLINIC NAME          '
BY INP_REFERAL AS 'REFERRAL, SOURCE' IN 73
BY VIS_PLACE AS '  PLACE OF VISIT      ' IN 49
IF CLINIC EQ &&CLN
IF VISIT_DATE FROM &&ST TO &&END
HEADING  CENTER
"REFERRAL SOURCE AND PLACE OF VISIT </1"
"FROM &&ST TO &&END"
END
JOIN CLEAR *
-RUN

```

```

-*****
-*
-* PROGRAM: CLINIC4.FEX          MODIFIED: 1/14/86 DRB *
-*
-* REPORT: 9                     MENU PROGRAM: RPTCRT5.FEX *
-*
-*****
-*
-SET &&RPT='JOB RELATED REPORT';
EX RPTCRT5
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&CLN=&CLN;
-CONT1
-SET &ST=EDIT(&&ST,'99/99/99');
-SET &END=EDIT(&&END,'99/99/99');
-RUN
JOIN DIAG1_CODE IN VISIT TO DGNCD IN DIAGNOS AS JOIN1
JOIN CLINIC IN VISIT TO CLINIC IN CLINIC AS JOIN2
TABLE FILE VISIT.JOBREL
  HEADING CENTER
  "</2 J O B   R E L A T E D   R E P O R T"
  "FOR &ST TO &END </2"
  PRINT PATIENT_ID AS 'PATIENT ID.' IN 35
  DIAG_DESCR AS '          DIAGNOSIS DESCRIPTION' IN 50
  BY CLINIC IN 25
  IF CLINIC EQ &&CLN
  IF VISIT_DATE FROM &&ST TO &&END
  IF JOBREL EQ Y
  ON CLINIC PAGE-BREAK
  END
JOIN CLEAR *
-RUN

```

```

-----
-*
-* PROCEDURE DEERS.FEX                                PREPARED BY D.R.B
-*                                                    ON 11/18/86
-* LOADS THE PATIENT DATA BASE FROM DEERS DATA
-* DATA IS IN FILE 'PATIENT.OUT'
-* RECORDS WITH DUPLICATE SSN NUMBERS ARE REJECTED
-----
SET MORE=OFF
FILEDEF RXXF200 DISK E: PATIENT.OUT
USE
D: PATIENT.FOC
END
MODIFY FILE PATIENT
TYPE AT START "PATIENT LOAD PROCEDURE IN PROGRESS"
FIXFORM PTID/C11 PTDOB/C6 CATEG/A3
FIXFORM PTLITHO/C8
FIXFORM REGDATE/C6 SEX/C1 RACE/C1 PREFIX/C2 PAYGRADE/C2
FIXFORM JOBCODE/C4 PTLOC/C6 TRAINEE/C1 VA/C1 HCI/C1
FIXFORM ZIP/C5 PTDUALSSN/C9 PTFORGN/C3
MATCH PTID
      ON NOMATCH INCLUDE
      ON MATCH REJECT
CHECK OFF
DATA ON RXXF200
END

```

```

-CRTCLEAR
-TYPE
-TYPE          DIAGNOS.FOC DUMP PROGRAM
-TYPE          PREPARE BERNOULLI CARTRIDGE FOR SHIPMENT TO FT. DETRICK
-TYPE
-TYPE          1. PLACE NEWLY FORMATED CARTRIDGE IN DRIVE E.
-TYPE
-TYPE          2. DRIVE D SHOULD CONTAIN DIAGNOS.FOC FILE (STANDARD).
-TYPE
-TYPE          HIT ENTER TO CONTINUE
-READ SYSIN
-CRTCLEAR
-TYPE
-TYPE          PREPARATION OF FT. DETRICK BERNOULLI DIAGNOSIS FILE
-TYPE
-TYPE          IN PROGRESS
-TYPE
USE C:SITE.IDF
END
FILEDEF SITEF DISK C:SITE.IDF
-RUN
-SET &&SITE='    ';
-READ SITEF &&SITE
USE
    D:DIAGNOS.FOC
    E:DETDIAG.FTM NEW
END
FILEDEF DETR DISK E:DETDIAG.FTM
-SET ALL=ON;
DEFINE FILE DIAGNOS
    SITE/A4='&&SITE';
END
-RUN
-TYPE DIAGNOSIS RECORDS BEING PROCESSED
TABLEF FILE DIAGNOS
PRINT SITE SEG.DGNCD
ON TABLE SAVE AS DETR
END
-RUN

```

DIAGNOS.MAS

```
FILENAME=DIAGNOS,SUFFIX=FOC
SEGNAME=DIAGNOS,SEGTYPE=S1
FIELDNAME=DIAG_CODE,    ALIAS=DGNCD,    FORMAT=A5,    FIELDTYPE=I,$
FIELDNAME=DIAG_DESCR,  ALIAS=DGNDSC,    FORMAT=A50    , $
END
DBA=LUGNUT,$
USER=NUTMEG, ACCESS=RW, $
```

```

*-----
*
* PROCEDURE LOADCLIN
*
* LOADS THE CLINIC DATA BASE
* DATA IS IN FILE 'BAMCOP.PRN'
*
*-----
FILEDEF LOADCLIN DISK A:BRAGOP.PRN
USE
D:CLINIC.FOC
END
-CRTCLEAR
MODIFY FILE CLINIC
FIXFORM X6 CLUCA/C4 CL_TITLE/C20
MATCH CLINIC
    ON NOMATCH INCLUDE
    ON MATCH CONTINUE
DATA ON LOADCLIN
END

```

```

*****
-* MAIN.FEX                      MAIN FOCUS DATA BASE MENU      *
-*                               *                                *
-*          MODIFIED 4/14/86   D R BOLLING                        *
-*          MODIFIED 5/7/86   DRB CORRECT RPT6 LABEL             *
-*          MODIFIED 12/16/86 DRB SYSTEM TO BE ON C DRV*         *
*****
SET PAUSE=OFF
SET COLOR=ON
SET BACK=BLUE,FORE=IWHITE
SET MSG=OFF
SET PRINT=OFFLINE
USE
D:PROCEDUR.FOC
D:DIAGNOS.FOC
D:CLINIC.FOC
D:PATIENT.FOC
D:PROVIDER.FOC
E:VISIT.FOC
END
-RUN
-SECTION1
-CRTCLEAR
-SET &MSG='  ';
-SET &REPLY='  ';
-MAIN
-CRTFORM
-"
-"          AMBULATORY CARE DATA MANAGEMENT SYSTEM
-"
-"          PLEASE SELECT A PROCESS
-"
-"  A ... Load Patient Registration          F ... File Maintenance
-"  B ... Load Provider Registration          G ... Tabletalk - Report S
-"  C ... Load Encounter Forms                H ... Rebuild Patient/Prov
-"  D ... Correct Data Base                    I ... Rebuild Encounter Fi
-"  E ... Prepare Ft. Detrick Shipment        J ... Standard Reports Opt
-"
-"          X .. LOGOFF
-"
-"          ENTER YOUR SELECTION ..... <.IRHW.&REPLY  <77 ||
-"          <.H.YELL.D.&MSG <77 ||"
-"
-SET &DEST=DECODE &REPLY (A PTLOAD B PRLOAD C ENLOAD D CORRECT
-      E SHIPMENT F MAINT G REPORT H PRREBLT I ENREBLT J SCRIT X LOGOFF
-      ELSE NOPE);
-GOTO &DEST
-PTLOAD

```

```

-CRTFORM
-" "
-"          PLACE CARTRIDGE WITH PATIENT.OUT FILE IN E DRIVE"
-" "
-"          HIT ENTER TO CONTINUE"
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
EX RXXF200
-RUN
-CRTFORM
-" "
-"          REPLACE CARTRIDGE WITH VISIT.FOC FILE IN E DRIVE"
-"          -----
-" "
-"          HIT ENTER TO CONTINUE"
-RUN
-GOTO SECTION1
-PRLOAD
-CRTFORM
-" "
-"          PLACE CARTRIDGE WITH PROVIDER.OUT FILE IN E DRIVE"
-" "
-"          HIT ENTER TO CONTINUE"
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
EX RXXF300
-RUN
-CRTFORM
-" "
-"          REPLACE CARTRIDGE WITH VISIT.FOC FILE IN E DRIVE"
-"          -----
-" "
-"          HIT ENTER TO CONTINUE"
-RUN
-GOTO SECTION1
-ENLOAD
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
EX RXXF400
-RUN
DOS ERASE D:VISIT.OUT
-RUN
-GOTO SECTION1
-MAINT
FILETALK
-RUN
-GOTO SECTION1
-REPORT
TABLETALK
-RUN
-GOTO SECTION1
-PRREBLT
EX RXXPTREB
-RUN
EX RXXPRREB

```

```

-RUN
-GOTO SECTION1
-ENREBLT
EX RXXENREB
-RUN
-GOTO SECTION1
-NOPE
-SET &MSG = 'INCORRECT SELECTION ENTERED. PLEASE TRY AGAIN';
-GOTO MAIN
-LOGOFF
-GOTO EXIT
-***** REPORT MENU *****
-SCRIT
-SET &MSG = ' ';
-SET &PRT = '1';
-SET &SEL = ' ';
-SET &&RPT = ' ';
-SET &STITLE = 'REPORT SELECTION MENU';
-SET &&NITE = 'N';
-*
-SECTION2
-CRTFORM
-"
-"          AMBULATORY CARE DATA MANAGEMENT SYSTEM
-"
-"          <.H.A.D.&STITLE <77 ||"
-"
-" 1 - ENCOUNTERS BY PROV. BY CLINIC   7 - NUMBER VISITS IN ALL CLINIC
-" 2 - PATIENT CATEGORY                 8 - PROVIDER DIAGNOSIS FOR ALL
-" 3 - CLINIC REFERRAL SOURCES          9 - JOB RELATED REPORT
-" 4 - PROVIDER DIAGNOSIS REPORT        10 - MILITARY UNIT REPORT
-" 5 - PROVIDER PROCEDURE REPORT        11 - HEALTH UNIT REPORT
-" 6 - TIME AND APPT STATUS             12 - CLINIC ACTIVITY REPORT
-" N - STACK NIGHT REPORTS             X - EXIT TO PREVIOUS MENU
-"
-"
-"          ENTER REPORT OPTION (1 = SCREEN 2 = PRINTER): <.IR'WT.&PRT <77
-"
-"          ENTER SELECTION: <.IRHW.&SEL <77 ||"
-"
-"          <.H.YELL.D.&MSG <77 ||"
-"
-SET &DES1=DECODE &PRT (1 ONIT 2 OFFIT ELSE BADSEL);
-GOTO &DES1
-GOODSEL
-SET &DEST=DECODE &SEL (1 RPT1 2 RPT2 3 RPT3 4 RPT4 5 RPT5 6 RPT6
- 7 RPT7 8 RPT8 9 RPT9 10 RPT10 11 RPT11 12 RPT12 N TONIGHT
- X SECTION1 ELSE BADSEL);
-GOTO &DEST
-BADSEL
-SET &MSG = 'INCORRECT SELECTION ENTERED. PLEASE TRY AGAIN';
-GOTO SECTION2

```

```

-*
-OFFIT
OFFLINE
SET PANEL=135
SET WIDTH=135
-GOTO GOODSEL
-ONIT
ONLINE
-GOTO GOODSEL
-*
-RPT1
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX CLINIC1
-RUN
-GOTO SCRIT
-*
-RPT2
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX CLINIC2
-RUN
-GOTO SCRIT
-*
-RPT3
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX CLINIC3
-RUN
-GOTO SCRIT
-*
-RPT4
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX PROV1
-RUN
-GOTO SCRIT
-*
-RPT5
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX PROV2
-RUN
-GOTO SCRIT
-*
-RPT6
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX PROV3
-RUN
-GOTO SCRIT
-*
-RPT7
-TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX PROV4
-RUN
-GOTO SCRIT
-*

```

```

-RPT8
- TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX PROV5
-RUN
-GOTO SCRIT
-*
-RPT9
- TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX CLINIC4
-RUN
-GOTO SCRIT
-RPT10
- TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX MILT1
-RUN
-GOTO SCRIT
-*
-RPT11
- TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX MILT2
-RUN
-GOTO SCRIT
-*
-RPT12
- TYPE "PLEASE WAIT - - PROGRAM LOADING"
  EX MILT3
-RUN
-GOTO SCRIT
-***** CORRECTION DATA BASE MENU *****
-CORRECT
-CRTCLEAR
-SET &REPLY=' ';
-SET &MSG=' ';
-SET &CTITLE='CORRECTION MENU';
-SECTION3
-CRTFORM
-"
-"
-"      AMBULATORY CARE DATA MANAGEMENT SYSTEM
-"
-"      <.H.A.D.&CTITLE <77 ||"
-"
-"      PLEASE SELECT A PROCESS
-"
-"      A ... Correct Clinic Code           E ... Future Application
-"      B ... Correct Provider Code         F ... Future Application
-"      C ... Correct Patient ID            G ... Future Application
-"      D ... Future Application             H ... Future Application
-"
-"      X ... RETURN TO PREVIOUS MENU

```

```

-"
-"      ENTER YOUR SELECTION ..... <.IRHW.&REPLY  <77 ||
-"      <.H.Y.D.&MSG <77 ||"
-"

```

```

-SET &DEST=DECODE &REPLY (A CLINIC B PROV C PATIEN D DSEL E ESEL F FSE
-      G GSEL H HSEL X SECTION1 ELSE NOGO);
-GOTO &DEST
-CLINIC
EX RXXFCCL
-RUN
-GOTO CORRECT
-PROV
EX RXXFCPR
-RUN
-GOTO CORRECT
-PATIEN
EX RXXFCPA
-RUN
-GOTO CORRECT
-DSEL
-GOTO CORRECT
-ESEL
-GOTO CORRECT
-FSEL
-GOTO CORRECT
-GSEL
-GOTO CORRECT
-HSEL
-GOTO CORRECT
-NOGO
-SET &MSG = 'INCORRECT SELECTION ENTERED.  PLEASE TRY AGAIN';
-GOTO SECTION3
-*****      PREPARE SHIPMENT TO DETRICK MENU      *****
-SHIPMENT
-SET &HTITLE='PREPARE SHIPMENT TO FT. DETRICK';
-SET &MSG= ' ';
-SECTION4
-CRTCLEAR
-SET &REPLY=' ';
-CRTFORM

```

```

-"
-"      AMBULATORY CARE DATA MANAGEMENT SYSTEM
-"
-"      <.H.A.D.&HTITLE <77 ||"
-"
-"      PLEASE SELECT A PROCESS
-"
-"      A ... VISIT File Preparation          E ... Future Application
-"      B ... PROVIDER File Preparation       F ... Future Application
-"      C ... PATIENT File Preparation        G ... Future Application

```

```

-"
-" D ... CLINIC File Preparation          H ... Future Application
-"
-"          X - EXIT TO PREVIOUS MENU
-"
-"          ENTER YOUR SELECTION ..... <.IRHW.&REPLY  <77 ||
-"          <.H.YELL.D.&MSG <77 ||"
-"
-SET &DEST=DECODE &REPLY (A VIDUMP B PRDUMP C PADUMP D CLDUMP E ESELS
-   F FSELS G GSELS H HSELS X SECTION1 ELSE NOGOS);
-GOTO &DEST
-NOGOS
-SET &MSG = 'INCORRECT SELECTION ENTERED.  PLEASE TRY AGAIN';
-GOTO SECTION4
-VIDUMP
EX RXXFDVI
-RUN
-GOTO SHIPMENT
-PRDUMP
EX RXXFDPR
-RUN
-GOTO SHIPMENT
-PADUMP
EX RXXFDPA
-RUN
-GOTO SHIPMENT
-CLDUMP
EX RXXFDCL
-RUN
-GOTO SHIPMENT
-ESELS
-GOTO SHIPMENT
-FSELS
-GOTO SHIPMENT
-GSELS
-GOTO SHIPMENT
-HSELS
-GOTO SHIPMENT
-*****      NIGHT REPORT PROCESSING      *****
-TONIGHT
-SET &&NITE='Y';
DOS IF EXIST C:STKEX.DAT ERASE C:STKEX.DAT
DOS IF EXIST E:NITE.DAT ERASE E:NITE.DAT
DOS FI STKEX DISK C:STKEX.DAT (RECFM FB LRECL 120
DOS FI STKEX DISK C:STKEX.DAT APPEND
OFFLINE CLOSE
FI OFFLINE DISK E:NITE.DAT
FI OFFLINE DISK E:NITE.DAT APPEND
-RUN
SET MORE=OFF
-SET &STKCTR = '0';
-SET &READEX = '

```

```

-SET &MSG = ' ';
-SET &NTITLE='REPORT SELECTION MENU - FOR NIGHT PROCESSING';
-SET &PRT = '1';
-SET &REPLY = ' ';
-SET &&RPT = ' ';
-*** SET REPORT FLAGS FOR SELECTION TRACKING ***
-SET &RF01 = ' ';
-SET &RF02 = ' ';
-SET &RF03 = ' ';
-SET &RF04 = ' ';
-SET &RF05 = ' ';
-SET &RF06 = ' ';
-SET &RF07 = ' ';
-SET &RF08 = ' ';
-SET &RF09 = ' ';
-SET &RF10 = ' ';
-SET &RF11 = ' ';
-SET &RF12 = ' ';
-SCRPSW
-*
-SECTION5
-CRTFORM
-"
-"      <.H.A.D.&NTITLE <77 ||"
-" <77 ||"
-" <D.&RF01      1 - ENCOUNTERS BY PROVIDER BY CLINIC <77 ||"
-" <D.&RF02      2 - PATIENT CATEGORY <77 ||"
-" <D.&RF03      3 - CLINIC REFERRAL SOURCES <77 ||"
-" <D.&RF04      4 - PROVIDER DIAGNOSIS REPORT <77 ||"
-" <D.&RF05      5 - PROVIDER PROCEDURE REPORT <77 ||"
-" <D.&RF06      6 - PROVIDER REPORT FOR TIME AND APPT STATUS <77 ||"
-" <D.&RF07      7 - PROVIDER REPORT FOR # VISITS IN ALL CLINICS <77 ||"
-" <D.&RF08      8 - PROVIDER DIAGNOSIS FOR ALL CLINICS <77 ||"
-" <D.&RF09      9 - JOB RELATED REPORT <77 ||"
-" <D.&RF10     10 - MILITARY UNIT REPORT <77 ||"
-" <D.&RF11     11 - HEALTH UNIT REPORT <77 ||"
-" <D.&RF12     12 - CLINIC ACTIVITY REPORT <77 ||"
-"      E - EXECUTE NIGHT REPORTS NOW      X - RETURN TO PREVIOUS MENU, CLEA
-"
-"      ENTER SELECTION: <.IRHW.&REPLY <77 ||"
-"      <.H.YELL.D.&MSG <77 ||"
-"
-SET &DEST=DECODE &REPLY (1 RTIME 2 RTIME 3 RTIME 4 RTIME 5 RTIME
-      6 RTIME 7 RTIME 8 RTIME 9 RTIME 10 RTIME 11 RTIME 12 RTIME
-      X EXITIT E EXECIT ELSE NOGON);
-GOTO &DEST
-NOGON
-SET &MSG = 'INCORRECT SELECTION ENTERED. PLEASE TRY AGAIN';
-GOTO SECTION5
-*
-RTIME
-SET &STKCTR = &STKCTR + 1;

```

```

-SET &DEST=DECODE &REPLY (1 RPT1N 2 RPT2N 3 RPT3N 4 RPT4N 5 RPT5N 6 RP
-      7 RPT7N 8 RPT8N 9 RPT9N 10 RPT10N 11 RPT11N 12 RPT12N);
-GOTO &DEST
-EXITIT
OFFLINE CLOSE
DOS FI OFFLINE PRINTER
-SET &&NITE = 'N';
-RUN
-GOTO SCRIT
-*****
-EXECIT
-CRTFORM
-" "
-" "
-" "
-" "
-" "
-" "
-" "
-      ARE YOU READY TO RUN NIGHT TIME PROCESSING "
-" "
-" "
-      ENTER Y (YES) OR N (NO) OR A (ABORT) :<&SEL"
-IF &SEL EQ 'Y' GOTO TESTSTK;
-IF &SEL EQ 'A' GOTO ABORTIT;
-SET &MSG = 'PLEASE ENTER ADDITION REPORTS NOW      ';
-GOTO SCRIT
-*
-ABORTIT
-SET &MSG = 'ABORT THE RUN --- TRY ENTERING AGAIN      ';
-GOTO START
-TESTSTK
-IF &STKCTR GT '0' GOTO RUNITNOW;
-SET &MSC = 'YOU HAVE NOT YET SELECTED ANY REPORTS      ';
-GOTO START
-*
-RUNITNOW
-*****
-*
-      E X E C U T E      N I G H T      P R O C E S S I N G
-*
-SET &&NITE = 'Z';
-DOS FI STKEX DISK C:STKEX.DAT
OFFLINE
SET PANEL = 132
SET WIDTH = 132
-RUN
EX STKEX.DAT
-TYPE
-TYPE *****
-TYPE *
-TYPE *

```

```

- TYPE *      ----          NIGHT REPORTS IN PROGRESS          ----
- TYPE *
- TYPE *      ! ! !      W A R N I N G      ! ! !
- TYPE *
- TYPE *      * * *      D O      N O T      D I S T U R B      C O M P U T E R      * *
- TYPE *
- TYPE *
- TYPE *      TOTAL REPORTS TO BE PROCESSED:          &STKCTR
- TYPE *
- TYPE *      *****
- RUN
- GOTO EXITIT
- *
- *****
- RPT1N
- *
EX RPTCRT
- RUN
- IF &&RTN EQ 'X' GOTO SECTION5;
- *
- WRITE STKEX EX CLINIC1 CLN=&&CLN, ST=&&ST, END=&&END, PROV=&&PROV
- SET &MSG = 'REPORT 1 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
- SET &RF01 = '*';
- GOTO SECTION5
- *****
- RPT2N
- *
EX RPTCRT
- RUN
- IF &&RTN EQ 'X' GOTO SECTION5;
- *
- WRITE STKEX EX CLINIC2 CLN=&&CLN, ST=&&ST, END=&&END, PROV=&&PROV
- SET &MSG = 'REPORT 2 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
- SET &RF02 = '*';
- GOTO SECTION5
- *****
- RPT3N
- *
EX RPTCRT
- RUN
- IF &&RTN EQ 'X' GOTO SECTION5;
- *
- WRITE STKEX EX CLINIC3 CLN=&&CLN, ST=&&ST, END=&&END, PROV=&&PROV
- SET &MSG = 'REPORT 3 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
- SET &RF03 = '*';
- GOTO SECTION5
- *****
- RPT4N
- *
EX RPTCRT
- RUN
- IF &&RTN EQ 'X' GOTO SECTION5;

```

```

- *
-WRITE STKEX EX PROV1 CLN=&&CLN, ST=&&ST, END=&&END, PROV=&&PROV
-SET &MSG = 'REPORT 4 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
-SET &RF04 = '*';
-GOTO SECTION5
-*****
-RPT5N
- *
EX RPTCRT
-RUN
-IF &&RTN EQ 'X' GOTO SECTION5;
- *
-WRITE STKEX EX PROV2 CLN=&&CLN, ST=&&ST, END=&&END, PROV=&&PROV
-SET &MSG = 'REPORT 5 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
-SET &RF05 = '*';
-GOTO SECTION5
-*****
-RPT6N
- *
EX RPTCRT
-RUN
-IF &&RTN EQ 'X' GOTO SECTION5;
- *
-WRITE STKEX EX PROV3 CLN=&&CLN, ST=&&ST, END=&&END, PROV=&&PROV
-SET &MSG = 'REPORT 6 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
-SET &RF06 = '*';
-GOTO SECTION5
-*****
-RPT7N
- *
EX RPTCRT1
-RUN
-IF &&RTN EQ 'X' GOTO SECTION5;
- *
-WRITE STKEX EX PROV4 ST=&&ST, END=&&END, PROV=&&PROV
-SET &MSG = 'REPORT 7 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
-SET &RF07 = '*';
-GOTO SECTION5
-*****
-RPT8N
- *
EX RPTCRT1
-RUN
-IF &&RTN EQ 'X' GOTO SECTION5;
- *
-WRITE STKEX EX PROV5 ST=&&ST, END=&&END, PROV=&&PROV
-SET &MSG = 'REPORT 8 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
-SET &RF08 = '*';
-GOTO SECTION5
-*****
-RPT9N

```

```

-*
EX RPTCRT
-RUN
-IF &&RTN EQ 'X' GOTO SECTION5;
-*
-WRITE STKEX EX CLINIC4 CLN=&&CLN, ST=&&ST, END=&&END, PROV=&&PROV
-SET &MSG = 'REPORT 9 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
-SET &RF09 = '*';
-GOTO SECTION5
-*****
-RPT10N
-*
EX RPTCRT2
-RUN
-IF &&RTN EQ 'X' GOTO SECTION5;
-*
-WRITE STKEX EX MILT1 CAT=&&CAT, ST=&&ST, END=&&END
-SET &MSG = 'REPORT 10 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
-SET &RF10 = '*';
-GOTO SECTION5
-*****
-RPT11N
-*
EX RPTCRT3
-RUN
-IF &&RTN EQ 'X' GOTO SECTION5;
-SET &COM = ',';
-*
-WRITE STKEX EX MILT2 FT=&&FT&COM
-WRITE STKEX LOC1=&&LOC1, LOC2=&&LOC2, LOC3=&&LOC3, LOC4=&&LOC4&COM
-WRITE STKEX ST=&&ST, END=&&END, &&CAT
-SET &MSG = 'REPORT 11 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
-SET &RF11 = '*';
-GOTO SECTION5
-*****
-RPT12N
-*
EX RPTCRT4
-RUN
-IF &&RTN EQ 'X' GOTO SECTION5;
-SET &COM = ',';
-*
-WRITE STKEX EX MILT3 CLN=&&CLN, ST=&&ST, END=&&END&COM
-WRITE STKEX RP=&&RP
-SET &MSG = 'REPORT 12 HAS BEEN STACKED FOR NIGHT TIME PROCESSING';
-SET &RF12 = '*';
-GOTO SECTION5
-EXIT

```

```

-*****
-*
-* PROGRAM: MILT1.FEX MODIFIED: 1/14/86 DRB *
-*
-* REPORT: 10 MENU PROGRAM: RPTCRT2.FEX *
-*
-*****
-*
-SET &&RPT = 'MILITARY UNIT REPORT';
EX RPTCRT2
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&CAT=&CAT;
-SET &&FT=&FT;
-CONT1
JOIN DIAG1_CODE IN VISIT TO DGNCD IN DIAGNOS AS DJOIN
JOIN PATIENT_ID IN VISIT TO PATIENT_ID IN PATIENT AS JJOIN
DEFINE FILE VISIT
MALE/I4=IF SEX EQ 'M' THEN 1 ELSE 0;
FEMALE/I4 = IF SEX EQ 'F' THEN 1 ELSE 0;
QUARTERS/I4=IF SUPPL_DISP EQ 'J' THEN 1 ELSE 0;
PROFILES/I4=IF SUPPL_DISP EQ 'M' THEN 1 ELSE 0;
YES/I4= IF JOB_RELATED EQ 'Y' THEN 1 ELSE 0;
NO/I4= IF JOB_RELATED EQ 'N' THEN 1 ELSE 0;
END
TABLE FILE VISIT.SUPPL_DISP
HEADING CENTER
" MILITARY UNIT REPORT FOR FORT &&FT"
"FROM &&ST TO &&END </2"
COUNT LITHO_CODE NOPRINT
SUM.FEMALE IN 50
SUM.MALE IN 60
SUM.QUARTERS IN 70 SUM.PROFILES IN 80
SUM.YES AS ' JOB,RELATED, YES' IN 90
SUM.NO AS ' JOB,RELATED, NO' IN 100
BY HIGHEST PT_LOC AS 'UIC ' IN 30
BY SUPPL_DISP NOPRINT
IF PT_CATEGORY EQ &&CAT
IF VISIT_DATE FROM &&ST TO &&END
IF SUPPL_DISP EQ 'M' OR 'J'
END
JOIN CLEAR *
-RUN

```

```

-*****
-*
-* PROGRAM: MILT2.FEX MODIFIED: 1/14/86 DRB *
-*
-* REPORT: 11 MENU PROGRAM: RPTCRT3.FEX *
-*
-*****
-*
-SET &&RPT = 'UNIT HEALTH REPORT';
EX RPTCRT3
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&LOC1=&LOC1;
-SET &&LOC2=&LOC2;
-SET &&LOC3=&LOC3;
-SET &&LOC4=&LOC4;
-SET &&CAT=&CAT;
-SET &&FT=&FT;
-CONT1
JOIN DIAG1_CODE IN VISIT TO DGNCDD IN DIAGNOS AS DJOIN
JOIN PATIENT_ID IN VISIT TO PATIENT_ID IN PATIENT AS JJOIN
DEFINE FILE VISIT
DES/A50=IF DIAG_DESCR EQ ' ' THEN 'N O D E S C R I P T I O N O N
ELSE DIAG_DESCR;
MALE/I4=IF SEX EQ 'M' THEN 1 ELSE 0;
FEMALE/I4=IF SEX EQ 'F' THEN 1 ELSE 0;
END
TABLE FILE VISIT
"<25 UNIT HEALTH REPORT FOR &&FT"
"<30 FROM &&ST TO &&END </2"
COUNT LITHO_CODE NOPRINT
SUM.FEMALE IN 72 SUM.MALE IN 82 ROW-TOTAL COLUMN-TOTAL
BY DES AS ' DIAGNOSIS DESCRIPTION ' IN 1
IF PT_CATEGORY EQ &&CAT
IF VISIT_DATE FROM &&ST TO &&END
IF PTLOC EQ &&LOC1 OR &&LOC2 OR &&LOC3 OR &&LOC4
END
JOIN CLEAR *
-RUN

```

```

*****
-*
-* PROGRAM: MILT3.FEX MODIFIED: 1/14/86 DRB *
-*
-* REPORT: 12 MENU PROGRAM: RPTCRT4.FEX *
-*
*****
-*
EX RPTCRT4
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&RP=&RP;
-SET &&CLN=&CLN;
-CONT1
JOIN DIAG1_CODE IN VISIT TO DGNCD IN DIAGNOS AS DJOIN
JOIN PATIENT_ID IN VISIT TO PATIENT_ID IN PATIENT AS JJOIN
DEFINE FILE VISIT
DOB/I6=YMD(PT_DOB,&YMD);
AGE/I2=DOB/365;
Z_1/I4 = IF AGE LT 0 THEN 1 ELSE 0;
ONE_4/I4 = IF AGE GE 1 AND AGE LE 4 THEN 1 ELSE 0;
FIVE_14/I4 = IF AGE GE 5 AND AGE LE 14 THEN 1 ELSE 0;
FIF_19/I4 = IF AGE GE 15 AND AGE LE 19 THEN 1 ELSE 0;
TW_39/I4 = IF AGE GE 20 AND AGE LE 39 THEN 1 ELSE 0;
FORT_64/I4 = IF AGE GE 40 AND AGE LE 64 THEN 1 ELSE 0;
SIX_GT/I4 = IF AGE GE 65 THEN 1 ELSE 0;
WD/A50=IF DIAG_DESCR EQ ' ' THEN 'N O D E S C R I P T I O N O N
ELSE DIAG_DESCR;
END
TABLE FILE VISIT.PLACE
HEADING CENTER
"&&RP"
"FOR &&ST TO &&END </2"
"<40 AGE GROUP </1"
IF VISIT_DATE FROM &&ST TO &&END
IF PLACE EQ '3'
IF PROC_CODE EQ '90720'
IF CLINIC EQ &&CLN
IF PT_DOB NE 0
SUM CNT.INP_REFERAL NOPRINT
Z_1 AS ' 0,TO,12,MO' IN 54
ONE_4 AS ' 1,TO, 4,YR' IN 60
FIVE_14 AS ' 5,TO,14,YR' IN 66
FIF_19 AS '15,TO,19,YR' IN 72
TW_39 AS '20,TO,39,YR' IN 78

```

```

        FORT_64    AS '40,TO,64,YR' IN 84
        SIX_GT     AS '65, &,UP'    IN 90 ROW-TOTAL AND COLUMN-TOTAL
BY WD           AS '                DIAGNOSIS DESCRIPTION          ' IN
ON WD SKIP-LINE
FOOTING
" <54 REFERRALS IN <70 <CNT.INP_REFERAL"
" <54 TELEPHONE VISIT <70 <CNT.PLACE"
" <54 IMMUNIZATIONS <70 <CNT.PROC_CODE"
END
-RUN
JOIN PROC_CODE IN VISIT TO PROC_CODE IN PROCEDUR JJOIN4
DEFINE FILE VISIT
DOB/I6=YMD(PT_DOB,&YMD);
AGE/I2=DOB/365;
Z_1/I4    = IF AGE LT 0 THEN 1 ELSE 0;
ONE_4/I4   = IF AGE GE 1 AND AGE LE 4 THEN 1 ELSE 0;
FIVE_14/I4 = IF AGE GE 5 AND AGE LE 14 THEN 1 ELSE 0;
FIF_19/I4  = IF AGE GE 15 AND AGE LE 19 THEN 1 ELSE 0;
TW_39/I4   = IF AGE GE 20 AND AGE LE 39 THEN 1 ELSE 0;
FORT_64/I4 = IF AGE GE 40 AND AGE LE 64 THEN 1 ELSE 0;
SIX_GT/I4  = IF AGE GE 65 THEN 1 ELSE 0;
WD/A67=IF PROC_DESCR EQ ' ' THEN 'N O   D E S C R I P T I O N   O N
        ELSE PROC_DESCR;
END
-RUN
TABLE FILE VISIT.PLACE
HEADING CENTER
"&&RP"
"FOR &&ST TO &&END </2"
"<40 AGE GROUP </1"
IF VISIT_DATE FROM &&ST TO &&END
IF PLACE EQ '3'
IF PROC_CODE EQ '90720'
IF CLINIC EQ &&CLN
IF PT_DOB NE 0
SUM CNT.INP_REFERAL NOPRINT
    Z_1      AS ' 0,TO,12,MO' IN 54
    ONE_4     AS ' 1,TO, 4,YR' IN 60
    FIVE_14   AS ' 5,TO,14,YR' IN 66
    FIF_19    AS '15,TO,19,YR' IN 72
    TW_39     AS '20,TO,39,YR' IN 78
    FORT_64   AS '40,TO,64,YR' IN 84
    SIX_GT    AS '65, &,UP'    IN 90 ROW-TOTAL AND COLUMN-TOTAL
BY WD        AS '                PROCEDURE DESCRIPTION          ' IN
ON WD SKIP-LINE
FOOTING
" <54 REFERRALS IN <70 <CNT.INP_REFERAL"
" <54 TELEPHONE VISIT <70 <CNT.PLACE"
" <54 IMMUNIZATIONS <70 <CNT.PROC_CODE"
END
-RUN
JOIN CLEAR *

```

-RUN

OTHER.MAS

FILENAME=PROCEDUR,SUFFIX=FOC

SEGNAME=OTHERCOD,SEGTYPE=S1

FIELDNAME=OTHER_CODE, ALIAS=OTHC,

FORMAT=A5, FIELDTYPE=I,\$

FIELDNAME=OTHER_DESCR, ALIAS=OTHDSC,

FORMAT=A67 , \$

END

DBA=LUGNUT,\$

USER=NUTMEG, ACCESS=RW, \$

PATIENT.MAS

```

FILE=PATIENT,SUFFIX=FOC
SEGNAME=PTSEG,SEGTYPE=S1
  FIELD=PATIENT_ID,PTID,A11,FIELDTYPE=I,$
  FIELD=PT_DOB,PTDOB,I6YMD,$
  FIELD=PT_CATEGORY,CATEG,A3,$
  FIELD=PT_LITHO,PTLITHO,A8,$
  FIELD=PT_REGDATE,REGDATE,I6YMD,$
  FIELD=PT_GENDER,SEX,A1,$
  FIELD=PT_RACE,RACE,A1,$
  FIELD=PT_PREFIX,PREFIX,A2,$
  FIELD=PT_PAY_GRADE,PAYGRADE,A2,$
  FIELD=PT_JOB_CODE,JOBCODE,A4,$
  FIELD=PT_LOCATION,PTLOC,A6,$
  FIELD=PT_TRN_TDY,TRAINEE,A1,$
  FIELD=PT_VAELIG,VA,A1,$
  FIELD=PT_HCI,HCI,A1,$
  FIELD=PT_ZIP_CODE,ZIP,A5,$
  FIELD=PT_DUAL_SSN,PTDUALSSN,A9,$
  FIELD=PT_FOREIGN,PTFORGN,A3,$
END
DBA=LUGNUT,$
USER=NUTMEG, ACCESS=RW, $

```

```

-CRTCLEAR
-TYPE                                PROCEDUR.FOC DUMP PROGRAM
-TYPE                                PREPARE BERNOULLI CARTRIDGE FOR SHIPMENT TO FT. DETRICK
-TYPE
-TYPE                                1. PLACE NEWLY FORMATED CARTRIDGE IN DRIVE E.
-TYPE
-TYPE                                2. DRIVE D SHOULD CONTAIN PROCEDUR.FOC FILE (STANDARD)
-TYPE
-TYPE                                HIT ENTER TO CONTINUE
-READ SYSIN
-CRTCLEAR
-TYPE
-TYPE                                PREPARATION OF FT. DETRICK BERNOULLI PROCEDURE FILE
-TYPE
-TYPE                                IN PROGRESS
-TYPE
USE C:SITE.IDF
END
FILEDEF SITEF DISK C:SITE.IDF
-RUN
-SET &&SITE='      ';
-READ SITEF &&SITE
USE
      D:PROCEDUR.FOC
      E:DETPROC.FTM NEW
END
FILEDEF DETR DISK E:DETPROC.FTM
-SET ALL=ON;
DEFINE FILE PROCEDUR
      SITE/A4='&&SITE';
END
-RUN
-TYPE PROCEDURE RECORDS BEING PROCESSED
TABLEF FILE PROCEDUR
PRINT SITE SEG.PRCCD
ON TABLE SAVE AS DETR
END
-RUN

```

PROCEDUR.MAS

```
FILENAME=PROCEDUR,SUFFIX=FOC
SEGNAME=PROCEDUR,SEGTYPE=S1
FIELDNAME=PROC_CODE, ALIAS=PRCCD, FORMAT=A5, FIELDTYPE=I,$
FIELDNAME=PROC_DESCR, ALIAS=PRCDSC, FORMAT=A67, $
END
DBA=LUGNUT,$
USER=NUTMEG, ACCESS=RW, $
```

PROFILE.FEX

SET MSG=OFF
SET PAUSE=OFF
SET WIDTH=132
SET LINES=66
SET PANEL=132
SET COLOR=ON
OFFLINE
-EXIT

```

-*****
-*
-*   PROGRAM: PROV1.FEX           MODIFIED: 6/17/87   DRB   *
-*   FOR NEW DESIGN               *
-*   REPORT: 4                   MENU PROGRAM: RPTCRT.FEX *
-*                               *
-*****
-*
-SET &&RPT = 'PROVIDER DIAGNOSIS REPORT';
EX RPTCRT
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&CLN=&CLN;
-SET &&PROV=&PROV;
-CONT1
JOIN DIAG1_CODE IN VISIT TO DIAG_CODE IN DIAGNOS AS DJOIN
JOIN CLINIC IN VISIT TO CLINIC IN CLINIC AS JJOIN
JOIN PROV_ID IN VISIT TO PROV_ID IN PROVIDER AS JJOIN1
DEFINE FILE VISIT
  FINT/A2=EDIT(PROV_FST_NME,'9')||'.';
  NM/A20= (PROV_LST_NME||', '||FINT);
END
-*****
-*   GENERATE REPORT FOR PRIMARY PROVIDERS
-*****
-*
TABLE FILE VISIT
COUNT L1THO_CODE
BY CLINIC BY CL_TITLE
BY NM
BY PROV_ID AS PID
COUNT L1THO_CODE
SUM.DIAG_DESCR
COMPUTE PA=100 * C2/C1;
BY CLINIC BY CL_TITLE
BY NM
BY PROV_ID
BY DIAG1_CODE
IF CLINIC EQ &&CLN
IF PROV_ID EQ &&PROV
IF VISIT_DATE FROM &&ST TO &&END
ON TABLE HOLD AS HOLD1
END
-RUN
TABLE FILE HOLD1

```

```

HEADING CENTER
"UCA : <CLINIC <CL_TITLE"
"FIRST PROVIDER PRIMARY DIAGNOSIS REPORT"
"FROM &&ST TO &&END"
" "
"PROVIDER : <NM ID: <PROV_ID"
" "
" "
BY CLINIC NOPRINT
BY CL_TITLE AS ' CLINIC NAME '
BY NM NOPRINT
BY PROV_ID AS 'PROV, ID'
BY HIGHEST E09 NOPRINT
PRINT DIAG_DESCR AS 'DESCRIPTION'
E07 AS 'DIAGNOSIS,COUNT'
PA AS 'PERCENT' COLUMN-TOTAL
ON PROV_ID SUBTOTAL AS 'TOTAL FOR'
ON PROV_ID PAGE-BREAK
FOOTING
" "
" "
"NOTE: THERE IS NO WAY TO DETERMINE WHETHER A DIAGNOSIS WAS DONE BY T
NUMBER"
" ONE PROVIDER OR THE NUMBER TWO PROVIDER FOR ANY ONE ENCOUNTER.
THEREFORE,"
" THE ABOVE DATA MAY REFLECT DIAGNOSES SPECIFIED BY EITHER PROVIDER
END
-RUN
JOIN CLEAR JJOIN1
JOIN PROV2 IN VISIT TO PROV_ID IN PROVIDER AS JJOIN1
-*****
-* GENERATE REPORT FOR SECONDARY PROVIDERS
-*****
DEFINE FILE VISIT
PROV_ID/A5=PROV2;
FINT/A2=EDIT(PROV_FST_NME,'9')||'.';
NM/A20= (PROV_LST_NME||', '||FINT);
END
-*
TABLE FILE VISIT
COUNT LITHO_CODE
BY CLINIC BY CL_TITLE
BY NM
BY PROV_ID AS PID
COUNT LITHO_CODE
SUM.DIAG_DESCR
COMPUTE PA=100 * C2/C1;
BY CLINIC BY CL_TITLE
BY NM
BY PROV_ID
BY DIAG1_CODE
IF PROV_ID NE ' '

```

```

IF CLINIC EQ &&CLN
IF PROV_ID EQ &&PROV
IF VISIT_DATE FROM &&ST TO &&END
ON TABLE HOLD AS HOLD1
END
-RUN
TABLE FILE HOLD1
HEADING CENTER
"UCA : <CLINIC <CL_TITLE"
"SECONDARY PROVIDER PRIMARY DIAGNOSIS REPORT"
"FROM &&ST TO &&END"
" "
"PROVIDER : <NM ID: <PROV_ID"
" "
" "
BY CLINIC NOPRINT
BY CL_TITLE AS ' CLINIC NAME '
BY NM NOPRINT
BY PROV_ID AS 'PROV, ID'
BY HIGHEST E09 NOPRINT
PRINT DIAG_DESCR AS 'DESCRIPTION'
E07 AS 'DIAGNOSIS,COUNT'
PA AS 'PERCENT' COLUMN-TOTAL
ON PROV_ID SUBTOTAL AS 'TOTAL FOR'
ON PROV_ID PAGE-BREAK
FOOTING
" "
" "
"NOTE: THERE IS NO WAY TO DETERMINE WHETHER A DIAGNOSIS WAS DONE BY T
NUMBER"
" ONE PROVIDER OR THE NUMBER TWO PROVIDER FOR ANY ONE ENCOUNTER.
THEREFORE,"
" THE ABOVE DATA MAY REFLECT DIAGNOSES SPECIFIED BY EITHER PROVIDER
END
JOIN CLEAR *
-RUN

```

```

*****
-*
-* PROGRAM: PROV2.FEX MODIFIED: 6/17/87 DRB *
-* FOR NEW DESIGN *
-* REPORT: 5 MENU PROGRAM: RPTCRT.FEX *
-*
*****
-*
-SET &&RPT = 'PROVIDER PROCEDURE REPORT';
EX RPTCRT
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&CLN=&CLN;
-SET &&PROV=&PROV;
-CONT1
-*
JOIN CLINIC IN VISIT TO CLINIC IN CLINIC AS CJOIN
JOIN PROV_ID IN VISIT TO PROV_ID IN PROVIDER AS PVJOIN
JOIN PROC_CODE IN VISIT TO PROC_CODE IN PROCEDUR AS PCJOIN
*****
-* PRIMARY PROVIDER REPORT
*****
DEFINE FILE VISIT
  FINT/A2=EDIT(PROV_FST_NME,'9')||'.';
  NM/A20=(PROV_LST_NME||', '||FINT);
  WPROC/A67=IF PRCDSC EQ ' ' THEN 'NO DESCRIPTION GIVEN' ELSE PRCDSC;
  WNO/I4=1;
END
-*
TABLE FILE VISIT
COUNT WNO
BY CLINIC BY CL_TITLE
BY NM
BY PROV_ID AS PID
COUNT LITHO_CODE
COMPUTE PA=100 * C2/C1;
BY CLINIC BY CL_TITLE
BY NM
BY PROV_ID
BY PROC_CODE
BY WPROC
IF PROV_ID EQ &&PROV
IF PRCODE EQ '1'
IF CLINIC EQ &&CLN
IF VISIT_DATE FROM &&ST TO &&END

```

```

ON TABLE HOLD
END
-RUN
-*****
TABLE FILE HOLD
HEADING
"<47 UCA : <CLINIC <CL_TITLE"
"<47 PRIMARY PROVIDER PROCEDURE REPORT"
"<53 FROM &&ST TO &&END"
" "
"<47 PROVIDER: <NM ID: <PROV_ID"
" "
" "
PRINT CL_TITLE AS ' CLINIC NAME '
PROC_CODE AS 'PROCEDURE, CODE'
WPROC AS ' PROCEDURE DESCRIPTION
LITHO_CODE AS 'PROCEDURE, COUNT'
PA AS 'PERCENT'
BY CLINIC NOPRINT
BY CL_TITLE NOPRINT
BY NM NOPRINT
BY PROV_ID NOPRINT
BY HIGHEST PA NOPRINT
BY LITHO_CODE NOPRINT
ON PROV_ID SUBTOTAL AS 'TOTAL FOR'
ON PROV_ID PAGE-BREAK
END
-RUN
-*****
-* SECONDARY PROVIDER REPORT
-*****
JOIN CLEAR PVJOIN
JOIN PROVIDER_2 IN VISIT TO PROV_ID IN PROVIDER AS PVJOIN
-RUN
-*
DEFINE FILE VISIT
PROV_ID/A5=PROV2;
FINT/A2=EDIT(PROV_FST_NME,'9')||'. ';
NM/A20=(PROV_LST_NME||', '||FINT);
WPROC/A67=IF PRCDSC EQ ' ' THEN 'NO DESCRIPTION GIVEN' ELSE PRCDSC;
WNO/I4=1;
END
-*
TABLE FILE VISIT
COUNT WNO
BY CLINIC BY CL_TITLE
BY NM
BY PROV_ID
COUNT LITHO_CODE
COMPUTE PA=100 * C2/C1;
BY CLINIC BY CL_TITLE
BY NM

```

```

BY PROV_ID
BY PROC_CODE
BY WPROC
IF PROV_ID NE ' '
IF PROV_ID EQ &&PROV
IF PRCODE EQ '2'
IF CLINIC EQ &&CLN
IF VISIT_DATE FROM &&ST TO &&END
ON TABLE HOLD
END
-RUN
-*****
TABLE FILE HOLD
HEADING
"<47 UCA : <CLINIC <CL_TITLE"
"<47 SECONDARY PROVIDER PROCEDURE REPORT"
"<53 FROM &&ST TO &&END"
" "
"<47 PROVIDER: <NM ID: <PROV_ID"
" "
" "
PRINT CL_TITLE AS ' CLINIC NAME '
PROC_CODE AS 'PROCEDURE, CODE'
WPROC AS ' PROCEDURE DESCRIPTION'
LITHO_CODE AS 'PROCEDURE, COUNT'
PA AS 'PERCENT'
BY CLINIC NOPRINT
BY CL_TITLE NOPRINT
BY NM NOPRINT
BY PROV_ID NOPRINT
BY HIGHEST PA NOPRINT
BY LITHO_CODE NOPRINT
ON PROV_ID SUBTOTAL AS 'TOTAL FOR'
ON PROV_ID PAGE-BREAK
END
JOIN CLEAR *
-RUN

```

```

-*****
-*
-* PROGRAM: PROV3.FEX MODIFIED: 1/14/86 DRB *
-*
-* REPORT: 6 MENU PROGRAM: RPTCRT.FEX *
-*
-*****
-*
-SET &&RPT = 'PROVIDER REPORT FOR TIME AND APPOINTMENT STATUS';
EX RPTCRT
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&CLN=&CLN;
-SET &&PROV=&PROV;
-CONT1
JOIN CLINIC IN VISIT TO CLINIC IN CLINIC AS JJOIN
JOIN PROV_ID IN VISIT TO PROV_ID IN PROVIDER AS JJOIN2
-INCLUDE DCODE1
-RUN
TABLE FILE VISIT
HEADING CENTER
"PROVIDER REPORT"
"FOR TIME AND APPOINTMENT STATUS"
"FOR <CL_TITLE"
"FOR PROVIDER <PROV_ID <100 *** <PROV_LST_NME ***"
"FROM &&ST TO &&END </2"
COUNT LITHO_CODE AS ' APPT, COUNT' IN 84 PV1_TIME AS 'TIME,COUNT' IN
BY CLINIC AS 'CLINIC' BY CL_TITLE AS ' CLINIC, NAME'
BY PROV_ID AS 'PROV, ID.'
BY PV1_TIME AS ' PROV1, TIME '
BY APT_STA AS ' APPT, STATUS' IN 65
IF CLINIC EQ &&CLN
IF VISIT_DATE FROM &&ST TO &&END
IF PROV_ID EQ &&PROV
ON PV1_TIME SKIP-LINE
ON PROV_ID SUBTOTAL AS 'PROV ID'
ON PROV_ID PAGE-BREAK
END
JOIN CLEAR *
-RUN

```

```

*****
-*
-* PROGRAM: PROV4.FEX MODIFIED: 1/14/86 DRB *
-*
-* REPORT: 7 MENU PROGRAM: RPTCRT1.FEX *
-*
*****
-SET &&RPT = 'NUMBER OF VISITS FOR PROVIDER IN ALL CLINICS';
EX RPTCRT1
-RUN
-IF &&RTN EQ 'X' GOTO RTN ELSE GOTO CONT;
-RTN
-EXIT
-CONT
-IF &&NITE EQ 'Z' THEN GOTO ASET ELSE GOTO CONT1;
-ASET
-SET &&ST=&ST;
-SET &&END=&END;
-SET &&PROV=&PROV;
-CONT1
JOIN CLINIC IN VISIT TO CLINIC IN CLINIC AS JJOIN
JOIN PROV_ID IN VISIT TO PROV_ID IN PROVIDER AS JJOIN1
TABLE FILE VISIT
HEADING CENTER
"NUMBER OF VISITS FOR <PROV_LST_NME IN ALL CLINICS "
"FROM &&ST TO &&END </2"
COUNT LITHO_CODE COLUMN-TOTAL AS 'VISIT,COUNT' IN 75
BY PROV_ID AS 'PROVIDER, ID.' IN 10
BY PROV_LST_NME AS 'PROVIDER, NAME' IN 20
BY CLINIC IN 35 BY CL_TITLE AS ' CLINIC NAME ' IN 45
IF PROV_ID EQ &&PROV
IF VISIT_DATE FROM &&ST TO &&END
ON PROV_ID PAGE-BREAK
END
JOIN CLEAR *
-RUN

```

PROVIDER.MAS

FILE=PROVIDER,SUFFIX=FOC
SEGNAME=PROVIDER,SEGTYPE=S1
FIELD=PROV_ID,PROVID,A5,FIELDTYPE=I,\$
FIELD=PROV_DATE,PROVDATE,I6YMD,\$
FIELD=PROV_LST_NME,PRLNME,A16,\$
FIELD=PROV_FST_NME,PRFNME,A13,\$
FIELD=PROV_INIT,PRINIT,A1,\$
FIELD=PROV_STAT,PRSTAT,A1,\$
FIELD=PROV_CATEG,PRCATEG,A2,\$
FIELD=PROV_PRPOS,PRPOS,A2,\$
FIELD=PROV_PAYGR,PRPAYGR,A2,\$
FIELD=PROV_JOBCODE,PRJCODE,A7,\$
FIELD=PROV_SSN,PRSSN,A9,\$
FIELD=PROV_LITHO,PRLTH,A8,\$
END
DBA=LUGNUT,\$
USER=NUTMEG, ACCESS=RW, \$

```

*****
-*                                     *
-*   PROGRAM: RPTCRT.FEX                D BOLLING *
-*                                     *
-*   FOR REPORTS: 1, 2, 3, 4, 5, 6      4/21/86 *
-*                                     *
*****
-START
SET ALL = ON
-IF &&NITE EQ 'Z' GOTO EXITIT;
-SET &MSG = IF &&NITE EQ 'Y' THEN
-   'SCREENING CRITERIA FOR NIGHT TIME PROCESSING' ELSE
-   ' ';
-*
-GOTO CLRIT
-SCR
-*
-CRTFORM
-"      <25 REPORT CRITERIA SELECTION"
-" "
-" "
-*"      <25 REPORT: <D.&&RPT"
-" "
-" "
-"      ENTER THE CLINIC CODE : <.IRHWT.&&CLN>"
-" "
-" "
-"      VISIT DATE FROM (YMD) : <.IRHWT.&&ST>"
-"      TO (YMD) : <.IRHWT.&&END>"
-" "
-" "
-"      ENTER A DESIRED PROVIDER CODE : <.IRHWT.&&PROV>"
-" "
-" "
-"ENTER 'X' TO RETURN TO PRIOR MENU: <.IRHW.&&RTN"
-" "
-" <.HYD.&MSG"
-IF &&RTN EQ 'X' THEN GOTO XEXIT;
-SET &&CLN = IF &&CLN EQ ' ' THEN '$$$$' ELSE (&&CLN);
-SET &&PROV = IF &&PROV EQ ' ' THEN '$$$$' ELSE (&&PROV);
-IF ( &&ST GE '840101' AND &&ST LE '991230' ) OR
-   &&ST EQ '$$$$$$' GOTO GOODST;
-SET &MSG = 'START DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-GOODST
-IF ( &&END GE '840101' AND &&END LE '991230' ) OR
-   &&END EQ '$$$$$$' GOTO GOODEND;
-SET &MSG = 'ENDING DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-GOODEND
-TYPE
-TYPE

```

```

- TYPE
- TYPE
- TYPE REPORT IS NOW PROCESSING OR NIGHT REPORT BEING STACKED
- TYPE
- TYPE SELECTION CRITERIA
- TYPE
- TYPE DATES : FROM &&ST TO &&END
- TYPE PROVIDER: &&PROV
- TYPE CLINIC : &&CLN
- TYPE
- TYPE
- TYPE
- TYPE &&RPT
- SET &&RTN='O';
- EXIT
- CLRIT
- SET &&ST = ' ';
- SET &&END = ' ';
- SET &&PROV = ' ';
- SET &&CLN = ' ';
- SET &&RTN= ' ';
- GOTO SCR
- XEXIT
- SET &&RTN='X';
- EXIT
- EXITIT
- EXIT

```

```

-*****
-*
-* PROGRAM: RPTCRT1.FEX D BOLLING *
-*
-* FOR REPORTS: 7, 8 4/21/86 *
-*
-*****
-START
SET ALL = ON
-IF &&NITE EQ 'Z' GOTO EXITIT;
-SET &MSG = IF &&NITE EQ 'Y' THEN
- 'SCREENING CRITERIA FOR NIGHT TIME PROCESSING' ELSE ' ';
-*
-GOTO CLRIT
-SCR
-*
-CRTFORM
-" <25 REPORT CRITERIA SELECTION"
-" "
-" "
-*" <25 REPORT: <D.&&RPT"
-" "
-" "
-" "
-" "
-" VISIT DATE FROM (YMD) : <.IRHWT.&&ST>"
-" TO (YMD) : <.IRHWT.&&END>"
-" "
-" "
-" ENTER A DESIRED PROVIDER CODE : <.IRHWT.&&PROV>"
-" "
-" "
-" ENTER 'X' TO RETURN TO PRIOR MENU: <.IRHWT.&&RTN"
-" "
-" <.HYD.&MSG"
-IF &&RTN EQ 'X' GOTO XEXIT;
-SET &&PROV = IF &&PROV EQ ' ' THEN '$$$$' ELSE (&&PROV);
-IF ( &&ST GE '840101' AND &&ST LE '991230' ) OR
- &&ST EQ '$$$$$$' GOTO GOODST;
-SET &MSG = 'START DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-GOODST
-IF ( &&END GE '840101' AND &&END LE '991230' ) OR
- &&END EQ '$$$$$$' GOTO GOODEND;
-SET &MSG = 'ENDING DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-GOODEND
-TYPE
-TYPE
-TYPE
-TYPE
-TYPE REPORT IS NOW PROCESSING OR NIGHT REPORT BEING STACKED

```

```

- TYPE
- TYPE
- TYPE
- TYPE      DATES      : FROM &&ST      TO &&END
- TYPE      PROVIDER: &&PROV
- TYPE
- TYPE
- TYPE
- TYPE      &&RPT
- EXIT
- *
- XEXIT
- EXIT
- *
- CLRIT
- SET &&RTN=' ' ;
- SET &&ST = ' ' ;
- SET &&END = ' ' ;
- SET &&PROV = ' ' ;
- SET &&CLN = ' ' ;
- GOTO SCR
- EXITIT
- EXIT

```

```

-*****
-*
-* PROGRAM: RPTCRT2.FEX          MODIFIED:  1/14/86  DRB  *
-*                               4/21/86  DRB  *
-* FOR REPORTS: 10              *
-*                               *
-*****
-START
SET ALL = ON
-*
-IF &&NITE EQ 'Z' GOTO EXITIT;
-SET &MSG = IF &&NITE EQ 'Y' THEN
- 'SCREENING CRITERIA FOR NIGHT TIME PROCESSING' ELSE ' ';
-*
-GOTO CLRIT
-SCR
-*
-CRTFORM
-"      <25 REPORT CRITERIA SELECTION"
-" "
-" "
-*"      <25 REPORT: <D.&&RPT"
-" "
-" "
-"      ENTER THE PATIENT CATEGORY : <.IRHWT.&&CAT>"
-" "
-" "
-"      VISIT DATE FROM (YMD) : <.IRHWT.&&ST>"
-"      TO (YMD) : <.IRHWT.&&END>"
-" "
-" "
-"      ENTER NAME OF SITE      FORT : <.IRHWT.&&FT>"
-" "
-" "
-"ENTER 'X' TO RETURN TO PRIOR MENU: <.IRHW.&&RTN"
-" "
-" <.HYD.&MSG"
-IF &&RTN EQ 'X' THEN GOTO XEXIT;
-SET &&CAT=IF &&CAT EQ ' ' THEN '$$$' ELSE &&CAT;
-IF ( &&ST GE '840101' AND &&ST LE '991230' ) GOTO GOODST;
-IF &&FT NE ' ' GOTO GOODST ELSE GOTO FTMSG;
-SET &MSG = 'START DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-FTMSG
-SET &MSG='PLEASE ENTER NAME OF THE FORT OR LOCATION';
-GOTO SCR
-GOODST
-IF ( &&END GE '840101' AND &&END LE '991230' ) OR
- &&END EQ '$$$$$$' GOTO GOODEND;
-SET &MSG = 'ENDING DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-GOODEND

```

```

- TYPE
- TYPE
- TYPE
- TYPE
- TYPE REPORT IS NOW PROCESSING OR NIGHT REPORT BEING STACKED
- TYPE
- TYPE SELECTION CRITERIA
- TYPE
- TYPE DATES : FROM &&ST TO &&END
- TYPE PATIENT CATEGORY: &&CAT
- TYPE FORT: &&FT
- TYPE
- TYPE
- TYPE
- TYPE &&RPT
- SET &&RTN='O';
- EXIT
- CLRIT
- SET &&ST = ' ';
- SET &&END = ' ';
- SET &&CAT = ' ';
- SET &&FT = ' ';
- SET &&RTN= ' ';
- GOTO SCR
- XEXIT
- SET &&RTN='X';
- EXIT
- EXITIT
- EXIT

```

```

*****
-*
-* PROGRAM: RPTCRT3.FEX D BOLLING
-*
-* FOR REPORTS: 11 4/21/86
-*
*****
-START
SET ALL = ON
-SET &&LOC1=' ' ;
-SET &&LOC2=' ' ;
-SET &&LOC3=' ' ;
-SET &&LOC4=' ' ;
-SET &UICM='ENTER UNIT ID CODE (UP TO 4) :';
-*
-IF &&NITE EQ 'Z' GOTO EXITIT;
-SET &MSG = IF &&NITE EQ 'Y' THEN
- 'SCREENING CRITERIA FOR NIGHT TIME PROCESSING' ELSE ' ';
-GOTO CLRIT
-SCR
-*
-CRTFORM
-" <25 REPORT CRITERIA SELECTION"
-" "
-" "
-" <25 REPORT: <D.&&RPT"
-" "
-" "
-" ENTER NAME OF UNIT : <.IRHWT.&&FT>"
-"<D.&UICM <.IRHWT.&&LOC1 <.IRHWT.&&LOC2 <.IRHWT.&&LOC3 <.IRHWT.&&LOC4
-" ENTER THE PATIENT CATEGORY : <.IRHWT.&&CAT>"
-" "
-" "
-" VISIT DATE FROM (YMD) : <.IRHWT.&&ST>"
-" TO (YMD) : <.IRHWT.&&END>"
-" "
-" "
-" "
-" ENTER 'X' TO EXIT : <.IRHW.&&RTN"
-" "
-" <.HYD.&MSG"
-IF &&RTN EQ 'X' THEN GOTO XEXIT;
-IF &&LOC1 EQ ' ' GOTO ST ELSE GOTO SKIP;
-ST
-SET &&LOC2 = '$$$$$$';
-SET &&LOC4 = '$$$$$$';
-SET &&LOC1 = '$$$$$$';
-SET &&LOC3 = '$$$$$$';
-SKIP
-IF &&CAT NE ' ' GOTO GOODST ELSE GOTO CATMSG;
-IF &&FT NE ' ' GOTO GOODST ELSE GOTO FTMSG;

```

```

-IF ( &&ST GE '840101' AND &&ST LE '991230' ) GOTO GOODST;
-SET &MSG = 'START DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-CATMSG
-SET &MSG='YOU MUST ENTER A PATIENT CATEGORY';
-GOTO SCR
-FTMSG
-SET &MSG='PLEASE ENTER NAME OF THE FORT OR LOCATION';
-GOTO SCR
-GOODST
-IF ( &&END GE '840101' AND &&END LE '991230' ) OR
-  &&END EQ '$$$$$$' GOTO GOODEND;
-SET &MSG = 'ENDING DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-GOODEND
-TYPE
-TYPE
-TYPE
-TYPE
-TYPE REPORT IS NOW PROCESSING OR NIGHT REPORT BEING STACKED
-TYPE
-TYPE SELECTION CRITERIA
-TYPE
-TYPE DATES : FROM &&ST TO &&END
-TYPE PATIENT CATEGORY: &&CAT
-TYPE FORT: &&FT
-TYPE
-TYPE
-TYPE
-TYPE &&RPT
-SET &&RTN='O';
-EXIT
-CLRIT
-SET &&LOC1=' ';
-SET &&LOC2=' ';
-SET &&LOC3=' ';
-SET &&LOC4=' ';
-SET &&ST = ' ';
-SET &&END = ' ';
-SET &&CAT = ' ';
-SET &&FT = ' ';
-SET &&RTN= ' ';
-GOTO SCR
-XEXIT
-SET &&RTN='X';
-EXIT
-EXITIT
-EXIT

```

```

*****
-*                                     *
-*   PROGRAM: RPTCRT4.FEX              D BOLLING   *
-*                                     *
-*   FOR REPORTS: 12                  4/21/86      *
-*                                     *
*****
-START
SET ALL = ON
-IF &&NITE EQ 'Z' GOTO EXITIT;
-SET &MSG = IF &&NITE EQ 'Y' THEN
- 'SCREENING CRITERIA FOR NIGHT PROCESSING' ELSE ' ';
-*
-GOTO CLRIT
-SCR
-*
-CRTFORM
-"      <25 REPORT CRITERIA SELECTION"
-" "
-" "
-" "
-" "
-" ENTER TITLE OF REPORT      : <.IRHWT.&&RP>"
-" ENTER CLINIC (UCA)        : <.IRHWT.&&CLN>"
-" "
-" "
-"      VISIT DATE FROM (YMD) : <.IRHWT.&&ST>"
-"                        TO (YMD) : <.IRHWT.&&END>"
-" "
-" "
-" "
-" "
-" ENTER 'X' TO EXIT          : <.IRHW.&&RTN"
-" "
-" <.HYD.&MSG"
-IF &&RTN EQ 'X' THEN GOTO XEXIT;
-SET &&CLN=IF &&CLN EQ ' ' THEN '$$$' ELSE &&CLN;
-IF ( &&ST GE '840101' AND &&ST LE '991230' ) GOTO GOODST;
-IF &&RP NE ' ' GOTO GOODST ELSE GOTO FTMSG;
-SET &MSG = 'START DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-FTMSG
-SET &MSG='PLEASE ENTER REPORT TITLE DESIRED';
-GOTO SCR
-GOODST
-IF ( &&END GE '840101' AND &&END LE '991230' ) OR
- &&END EQ '$$$$$$' GOTO GOODEND;
-SET &MSG = 'ENDING DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-GOODEND
-TYPE
-TYPE

```

```

- TYPE
- TYPE
- TYPE      REPORT IS NOW PROCESSING OR NIGHT REPORT BEING STACKED
- TYPE
- TYPE      SELECTION CRITERIA
- TYPE
- TYPE      DATES      : FROM &&ST      TO &&END
- TYPE      CLINIC     : &&CLN
- TYPE
- TYPE
- TYPE
- TYPE
- TYPE      &&RP
- SET &&RTN='O';
- EXIT
- CLRIT
- SET &&ST = '      ';
- SET &&CLN= '      ';
- SET &&END = '      ';
- SET &&RP = '      ';
- SET &&RTN= '      ';
- GOTO SCR
- XEXIT
- SET &&RTN='X';
- EXIT
- EXITIT
- EXIT

```

```

-*****
-*
-* PROGRAM: RPTCRT5.FEX WRITTEN: 1/14/86 DRB *
-*
-* FOR REPORTS: 9 MODIFIED 4/21/86 DRB *
-*
-*****
-*
-START
SET ALL = ON
-IF &&NITE EQ 'Z' GOTO EXITIT;
-SET &MSG = IF &&NITE EQ 'Y' THEN
- 'SCREENING CRITERIA FOR NIGHT TIME PROCESSING' ELSE
- ' ';
-*
-*
-*
-GOTO CLRIT
-SCR
-*
-CRTFORM
-" <25 REPORT CRITERIA SELECTION"
-" "
-" "
-*" <25 REPORT: <D.&&RPT"
-" "
-" "
-" ENTER THE CLINIC CODE : <.IRHWT.&&CLN>"
-" "
-" "
-" VISIT DATE FROM (YMD) : <.IRHWT.&&ST>"
-" TO (YMD) : <.IRHWT.&&END>"
-" "
-" "
-" "
-" "
-"ENTER 'X' TO RETURN TO PRIOR MENU: <.IRHW.&&RTN"
-" "
-" <.HYD.&MSG"
-IF &&RTN EQ 'X' THEN GOTO XEXIT;
-SET &&CLN = IF &&CLN EQ ' ' THEN '$$$$' ELSE (&&CLN);
-IF ( &&ST GE '840101' AND &&ST LE '991230' ) OR
- &&ST EQ '$$$$$$' GOTO GOODST;
-SET &MSG = 'START DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-GOODST
-IF ( &&END GE '840101' AND &&END LE '991230' ) OR
- &&END EQ '$$$$$$' GOTO GOODEND;
-SET &MSG = 'ENDING DATE IS INVALID - PLEASE ENTER IN YYMMDD FORMAT';
-GOTO SCR
-GOODEND
-TYPE

```

```

- TYPE
- TYPE
- TYPE
- TYPE REPORT IS NOW PROCESSING OR NIGHT REPORT BEING STACKED
- TYPE
- TYPE SELECTION CRITERIA
- TYPE
- TYPE DATES : FROM &&ST TO &&END
- TYPE CLINIC : &&CLN
- TYPE
- TYPE
- TYPE &&RPT
- SET &&RTN='O';
- EXIT
- CLRIT
- SET &&ST = ' ';
- SET &&END = ' ';
- SET &&PROV = ' ';
- SET &&CLN = ' ';
- SET &&RTN= ' ';
- GOTO SCR
- XEXIT
- SET &&RTN='X';
- EXIT
- EXITIT
- EXIT

```

```

-*****
-*  RXXCCLD.FEX          D R BOLLING          05/06/87      *
-*                                                                *
-*  DUMPS CLINIC SEGMENTS TO FLAT FILE FOR CORRECTION      *
-*  REVISED FOR NEW DESIGN FORMS                          *
-*  CALLED FROM RXXFCCL.FEX                                *
-*****
-CRTCLEAR
END
USE
      E:VISIT.FOC
END
-SET ALL=ON;
-SET &&RECTYPE='1';
DEFINE FILE VISIT
      RECTP/A1='&&RECTYPE';
      CLINEW/A4='&&CLNEW';
END
-RUN
-TYPE RECORDS TYPE 1 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM
TABLEF FILE VISIT.TIME1
PRINT RECTP CLINEW VDATE PROV1 PTID SEG.LITHO
IF CLINIC EQ &&CLOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='2';
DEFINE FILE VISIT ADD
      RECTP/A1='&&RECTYPE';
      CLINEW/A4='&&CLNEW';
      PAD/A30='.....';
END
-RUN
-TYPE RECORDS TYPE 2 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.PCODE
PRINT RECTP CLINEW VDATE PROV1 PTID LITHO PCODE PRCODE PAD
IF CLINIC EQ &&CLOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='3';
DEFINE FILE VISIT ADD
      RECTP/A1='&&RECTYPE';
      CLINEW/A4='&&CLNEW';
      PAD/A35='.....';
END
-RUN
-TYPE RECORDS TYPE 3 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.SPCC

```

```

PRINT RECTP CLINEW VDATE PROV1 PTID LITHO SPCC PAD
IF CLINIC EQ &&CLOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='4';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    CLINEW/A4='&&CLNEW';
    PAD/A31='.....';
END
-RUN
- TYPE RECORDS TYPE 4 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.OTHER
PRINT RECTP CLINEW VDATE PROV1 PTID LITHO OTHER PAD
IF CLINIC EQ &&CLOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='5';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    CLINEW/A4='&&CLNEW';
    PAD/A2='..';
END
-RUN
- TYPE RECORDS TYPE 5 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.UIC
PRINT RECTP CLINEW VDATE PROV1 PTID LITHO SEG.UIC PAD
IF CLINIC EQ &&CLOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='6';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    CLINEW/A4='&&CLNEW';
    PAD/A35='.....';
END
-RUN
- TYPE RECORDS TYPE 6 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.SPROG
PRINT RECTP CLINEW VDATE PROV1 PTID LITHO SPROG PAD
IF CLINIC EQ &&CLOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='7';
DEFINE FILE VISIT ADD

```

```

RECTP/A1='&&RECTYPE';
CLINEW/A4='&&CLNEW';
PAD/A31='.....';
END
-RUN
- TYPE RECORDS TYPE 7 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.DX2CODE
PRINT RECTP CLINEW VDATE PROV1 PTID LITHO DIAG2_CODE PAD
IF CLINIC EQ &&CLOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='8';
DEFINE FILE VISIT ADD
  RECTP/A1='&&RECTYPE';
  CLINEW/A4='&&CLNEW';
  PAD/A35='.....';
END
-RUN
- TYPE RECORDS TYPE 8 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.ASSESMNT
PRINT RECTP CLINEW VDATE PROV1 PTID LITHO ASSESMNT PAD
IF CLINIC EQ &&CLOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='9';
DEFINE FILE VISIT ADD
  RECTP/A1='&&RECTYPE';
  CLINEW/A4='&&CLNEW';
  PAD/A34='.....';
END
-RUN
- TYPE RECORDS TYPE 9 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.DISP
PRINT RECTP CLINEW VDATE PROV1 PTID LITHO DISP PAD
IF CLINIC EQ &&CLOLD
ON TABLE SAVE AS HOLD
END
-RUN

```

```

*****
-* RXXCORL.FEX                D R BOLLING                05/06/87
-*
-* LOADS CORRECTED HOLD FILE BACK TO VISIT FILE
-* REVISED FOR NEW FORM DESIGN
-* CALLED FROM RXXFCCL.FEX OR RXXFCPR.FEX OR RXXFCPA.FEX
-*
*****
FILEDEF MODIF DISK C:HOLD.FTM
FILEDEF VISITLG DISK C:VISIT.LOG
-*
-START
DOS STATE E:VISIT.FOC
-RUN
-IF &RETCODE NE 0 GOTO UNEW ;
USE E:VISIT.FOC
END
-GOTO CONT
-UNEW
USE E:VISIT.FOC NEW
END
CREATE FILE VISIT
-CONT
-CRTCLEAR
MODIFY FILE VISIT
-*
LOG DUPL MSG OFF
LOG NOMATCH MSG OFF
TYPE AT START ON VISITLG " CORRECTION ERROR LOG      September 10, 1988
TYPE AT START "LOADING CORRECTED DATA      September 10, 1988  "
CHECK OFF
COMPUTE
    RECTYPE/A1 =;
FIXFORM RECTYPE/A1 CLINIC/C4 VDATE/C6 PROV1/C5
FIXFORM PTID/C11 LITHO/C8 X36
-----
-*                      LOAD OF THE VISIT KEY DATA FOR ALL FORMS.
-----
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT
MATCH VDATE
    ON MATCH CONTINUE
    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT
MATCH PROV1
    ON MATCH CONTINUE
    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT
MATCH PTID
    ON MATCH CONTINUE

```

```

ON NOMATCH INCLUDE
ON NOMATCH GOTO TESTRECT
MATCH LITHO
ON MATCH GOTO TESTRECT
ON NOMATCH INCLUDE
ON NOMATCH GOTO TESTRECT

```

-*

CASE TESTRECT

COMPUTE

CLINIC = CLINIC;

VDATE = VDATE;

PROV1 = PROV1;

PTID = PTID;

LITHO = LITHO;

IF RECTYPE EQ '1' GOTO PROC1TP;

IF RECTYPE EQ '2' GOTO PROC2TP;

IF RECTYPE EQ '3' GOTO PROC3TP;

IF RECTYPE EQ '4' GOTO PROC4TP;

IF RECTYPE EQ '5' GOTO PROC5TP;

IF RECTYPE EQ '6' GOTO PROC6TP;

IF RECTYPE EQ '7' GOTO PROC7TP;

IF RECTYPE EQ '8' GOTO PROC8TP;

IF RECTYPE EQ '9' GOTO PROC9TP;

ENDCASE

-*

CASE PROC1TP

-*

-*

COMMON FORM

-*

RECORD TRANSACTION TYPE '1' PROCESSING BEGINS

-*

FIXFORM X-36 FNUM/2 VCNT/1 TIME1/3 PROV2/5 TIME2/3 PROV2RES/1

FIXFORM APPSTAT/1 PLACE/1 REFERRAL/4 JOBREL/1 DUTY/1 QTR/1

FIXFORM PROF/1 NAVAIL/1 ADMIT/1 ILLN/1 INJ/1 PURP/1

FIXFORM RULE/1 DX1CODE/5

MATCH CLINIC

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH VDATE

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH PROV1

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH PTID

ON MATCH CONTINUE

ON NOMATCH REJECT

-*

MATCH LITHO

ON MATCH

UPDATE FNUM VCNT TIME1 PROV2 TIME2 PROV2RES APPSTAT

UPDATE PLACE REFERRAL JOBREL DUTY QTR PROF NAVAIL ADMIT

UPDATE ILLN INJ PURP RULE DX1CODE

```

    ON NOMATCH REJECT
ENDCASE
-----*
CASE PROC2TP
FIXFORM X-36
FIXFORM PCODE/5 PRCODE/1
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH VDATE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PROV1
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PTID
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH LITHO
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PCODE
    ON MATCH REJECT
    ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-----*

```

```

CASE PROC3TP
FIXFORM X-36
FIXFORM SPCC/1
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH VDATE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PROV1
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PTID
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH LITHO
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH SPCC
    ON MATCH REJECT
    ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-----*

```

```

CASE PROC4TP

```

```

FIXFORM X-36
FIXFORM OTHER/5
MATCH CLINIC
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH VDATE
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PROV1
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PTID
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH LITHO
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH OTHER
  ON MATCH REJECT
  ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

-----*

```

CASE PROC5TP
FIXFORM X-36
FIXFORM UIC/6  TIMPR1/3 TIMPR2/3 TIMTR1/3 TIMTR2/3 MACT/3 OACT/3
FIXFORM RMIL/3 DEPD/3  CIVIL/3  CSHEET/1
MATCH CLINIC
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH VDATE
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PROV1
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PTID
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH LITHO
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH UIC
  ON MATCH REJECT
  ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

-----*

```

CASE PROC6TP
FIXFORM X-36
FIXFORM SPROG/1
MATCH CLINIC

```

```

      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH VDATE
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PROV1
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PTID
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH LITHO
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH SPROG
      ON      MATCH REJECT
      ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

```

CASE PROC7TP
FIXFORM X-36 DX2CODE/5
MATCH CLINIC
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH VDATE
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PROV1
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PTID
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH LITHO
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH DX2CODE
      ON      MATCH REJECT
      ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

```

CASE PROC8TP
FIXFORM X-36 ASSESMNT/1
MATCH CLINIC
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH VDATE
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PROV1

```

```

      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH PTID
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH LITHO
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH ASSESMNT
      ON   MATCH REJECT
      ON  NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

```

CASE PROC9TP
FIXFORM X-36 DISP/2
MATCH CLINIC
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH VDATE
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH PROV1
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH PTID
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH LITHO
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH DISP
      ON   MATCH REJECT
      ON  NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

```

DATA ON MODIF
END

```

```

-*****
-*  RXPCAD.FEX          D R BOLLING          05/06/87      *
-*                                                              *
-*  DUMPS PATIENT  SEGMENTS TO FLAT FILE FOR CORRECTION  *
-*  REVISED FOR NEW DESIGN FORMS                        *
-*  CALLED FROM RXPCPA.FEX                               *
-*****
-CRTCLEAR
END
USE
    E:VISIT.FOC
END
-SET ALL=ON;
-SET &&RECTYPE='1';
DEFINE FILE VISIT
    RECTP/A1='&&RECTYPE';
    PATNEW/A11='&&PANEW';
END
-RUN
- TYPE RECORDS TYPE 1 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM
TABLEF FILE VISIT.LITHO
PRINT RECTP CLINIC VDATE PROV1 PATNEW SEG.LITHO
IF PTID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='2';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PATNEW/A11='&&PANEW';
    PAD/A30='.....';
END
-RUN
- TYPE RECORDS TYPE 2 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.PROC_CODE
PRINT RECTP CLINIC VDATE PROV1 PATNEW LITHO PCODE PCODE PAD
IF PTID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='3';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PATNEW/A11='&&PANEW';
    PAD/A35='.....';
END
-RUN
- TYPE RECORDS TYPE 3 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.SPCC

```

```

PRINT RECTP CLINIC VDATE PROV1 PATNEW LITHO SPCC PAD
IF PTID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='4';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PATNEW/A11='&&PANEW';
    PAD/A31='.....';
END
-RUN
-TYPE RECORDS TYPE 4 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.OTHER
PRINT RECTP CLINIC VDATE PROV1 PATNEW LITHO OTHER PAD
IF PTID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='5';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PATNEW/A11='&&PANEW';
    PAD/A2='..';
END
-RUN
-TYPE RECORDS TYPE 5 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.UIC
PRINT RECTP CLINIC VDATE PROV1 PATNEW LITHO SEG.UIC PAD
IF PTID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='6';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PATNEW/A11='&&PANEW';
    PAD/A35='.....';
END
-RUN
-TYPE RECORDS TYPE 6 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.SPECIAL_PROG
PRINT RECTP CLINIC VDATE PROV1 PATNEW LITHO SPROG PAD
IF PTID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='7';
DEFINE FILE VISIT ADD

```

```

RECTP/A1='&&RECTYPE';
PATNEW/A11='&&PANEW';
PAD/A31='.....';
END
-RUN
-TYPE RECORDS TYPE 7 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.DIAG2_CODE
PRINT RECTP CLINIC VDATE PROV1 PATNEW LITHO DIAG2_CODE PAD
IF PTID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='8';
DEFINE FILE VISIT ADD
  RECTP/A1='&&RECTYPE';
  PATNEW/A11='&&PANEW';
  PAD/A35='.....';
END
-RUN
-TYPE RECORDS TYPE 8 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.ASSESMNT
PRINT RECTP CLINIC VDATE PROV1 PATNEW LITHO ASSESMNT PAD
IF PTID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='9';
DEFINE FILE VISIT ADD
  RECTP/A1='&&RECTYPE';
  PATNEW/A11='&&PANEW';
  PAD/A34='.....';
END
-RUN
-TYPE RECORDS TYPE 9 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.DISP
PRINT RECTP CLINIC VDATE PROV1 PATNEW LITHO DISP PAD
IF PTID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN

```

```

-*****
-*   RXXCPRD.FEX           D R BOLLING           05/06/87   *
-*                                                                *
-*   DUMPS PROVIDER SEGMENTS TO FLAT FILE FOR CORRECTION *
-*   REVISED FOR NEW DESIGN FORMS                      *
-*   CALLED FROM RXXFCPR.FEX                          *
-*****
-CRTCLEAR
END
USE
    E:VISIT.FOC
END
-SET ALL=ON;
-SET &&RECTYPE='1';
DEFINE FILE VISIT
    RECTP/A1='&&RECTYPE';
    PRNEW/A5='&&PRNEW';
END
-RUN
-TYPE RECORDS TYPE 1 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM
TABLEF FILE VISIT.TIME1
PRINT RECTP CLINIC VDATE PRNEW PTID SEG.LITHO
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='2';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PRNEW/A5='&&PRNEW';
    PAD/A30='.....';
END
-RUN
-TYPE RECORDS TYPE 2 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.PCODE
PRINT RECTP CLINIC VDATE PRNEW PTID LITHO PCODE PRCODE PAD
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='3';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PRNEW/A5='&&PRNEW';
    PAD/A35='.....';
END
-RUN
-TYPE RECORDS TYPE 3 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.SPCC

```

```

PRINT RECTP CLINIC VDATE PRNEW PTID LITHO SPCC PAD
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='4';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PRNEW/A5='&&PRNEW';
    PAD/A31='.....';
END
-RUN
-TYPE RECORDS TYPE 4 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.OTHER
PRINT RECTP CLINIC VDATE PRNEW PTID LITHO OTHER PAD
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='5';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PRNEW/A5='&&PRNEW';
    PAD/A2='..';
END
-RUN
-TYPE RECORDS TYPE 5 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.UIC
PRINT RECTP CLINIC VDATE PRNEW PTID LITHO SEG.UIC PAD
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='6';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PRNEW/A5='&&PRNEW';
    PAD/A35='.....';
END
-RUN
-TYPE RECORDS TYPE 6 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.SPROG
PRINT RECTP CLINIC VDATE PRNEW PTID LITHO SPROG PAD
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='7';
DEFINE FILE VISIT ADD

```

```

RECTP/A1='&&RECTYPE';
PRNEW/A5='&&PRNEW';
PAD/A31='.....';
END
-RUN
- TYPE RECORDS TYPE 7 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.DX2CODE
PRINT RECTP CLINIC VDATE PRNEW PTID LITHO DIAG2_CODE PAD
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='8';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PRNEW/A5='&&PRNEW';
    PAD/A35='.....';
END
-RUN
- TYPE RECORDS TYPE 8 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.ASSESMNT
PRINT RECTP CLINIC VDATE PRNEW PTID LITHO ASSESMNT PAD
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN
-SET &&RECTYPE='9';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
    PRNEW/A5='&&PRNEW';
    PAD/A34='.....';
END
-RUN
- TYPE RECORDS TYPE 9 BEING PROCESSED
FILEDEF HOLD DISK C:HOLD.FTM APPEND
TABLEF FILE VISIT.DISP
PRINT RECTP CLINIC VDATE PRNEW PTID LITHO DISP PAD
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN

```

```

--*-----*
--*
--* PROCEDURE RXXF200.FEX                                PREPARED BY J.J.S
--*                                                    ON 06/04/84
--* LOADS THE PATIENT DATA BASE
--* DATA IS IN FILE 'PATIENT.OUT'
--* 29 OCT 86 CHANGE INPUT FILE TO E DRIVE    D BOLLING
--*-----*

SET PRINT=OFFLINE
FILEDEF RXXF200 DISK E: PATIENT.OUT
FILEDEF PATLOG DISK PATIENT.LOG
USE
D: PATIENT.FOC
END
-CRTCLEAR
MODIFY FILE PATIENT
LOG DUPL MSG OFF
LOG NOMATCH MSG OFF
TYPE AT START "PATIENT LOAD PROCEDURE IN PROGRESS"
COMPUTE
    RECTYPE/A3 =;
    FIXFORM RECTYPE/A3
    FIXFORM PTID/C11   PTDOB/C6   CATEG/A3
    FIXFORM PTLITHO/C8
    FIXFORM REGDATE/C6 SEX/C1 RACE/C1 PREFIX/C2 PAYGRADE/C2
    FIXFORM JOBCODE/C4 PTLOC/C6 TRAINEE/C1 VA/C1 HCI/C1
    FIXFORM ZIP/C5   PTDUALSSN/C9 PTFORGN/C3
    IF RECTYPE EQ 'XXX' GOTO TOP;
    IF RECTYPE EQ '200' GOTO LOADER;
    TYPE ON PATLOG "UNRECOGNIZED RECORD TYPE <RECTYPE FOR PATIENT <PTID"
CASE LOADER
MATCH PTID
    ON NOMATCH INCLUDE
    ON MATCH GOTO SEG2
ENDCASE
--*
CASE SEG2
MATCH PTLITHO
    ON NOMATCH TYPE ON PATLOG
        "PATIENT <PTID ON FILE .. NO MATCH FOR LITHO <PTLITHO"
    ON NOMATCH TYPE ON PATLOG
        "
        ... <D.PTLITHO"
    ON NOMATCH REJECT
    ON MATCH UPDATE CATEG REGDATE SEX RACE PREFIX PAYGRADE
    ON MATCH UPDATE JOBCODE PTLOC TRAINEE VA HCI ZIP PTDUALSSN PTFO
ENDCASE
CHECK OFF
DATA ON RXXF200
END

```

```

--*-----*
--*
--* PROCEDURE RXXF300.FEX                                PREPARED BY J.J.S
--*                                                    ON 06/04/84
--* LOADS THE PROVIDER DATA BASE
--* DATA IS IN FILE 'PPROVIDER.OUT'
--* 29 OCT 86 CHANGE INPUT FILE TO E DRIVE
--*-----*

FILEDEF RXXF300 DISK E:PROVIDER.OUT
FILEDEF PROVIDLG DISK PROVIDER.LOG
USE
D:PROVIDER.FOC
END
-CRTCLEAR
MODIFY FILE PROVIDER
LOG DUPL MSG OFF
LOG NOMATCH MSG OFF
TYPE AT START "PROVIDER LOAD PROCEDURE IN PROGRESS"
COMPUTE
    RECTYPE/A3 =;
COMPUTE
    TOR/A1 =;
FIXFORM RECTYPE/A3 TOR/A1
FIXFORM PROVID/5    PROVDATA/6    PRCATEG/2
FIXFORM PRFNME/13   PRLNME/16     PRINIT/1
FIXFORM PRSSN/9     PRSTAT/1      PRPOS/2
FIXFORM PRPAYGR/2   PRJCODE/7     PRLTH/8
IF RECTYPE EQ 'XXX' GOTO TOP;
IF RECTYPE EQ '300' GOTO LOADER;
TYPE "UNRECOGNIZED RECORD <RECTYPE FOR PROVIDER <PROVID"
CASE LOADER
IF TOR EQ '1' GOTO ADD1;
IF TOR EQ '2' GOTO CHANGE1;
IF TOR EQ '3' GOTO DELETE1;
TYPE "UNRECOGNIZED RECORD TYPE <TOR FOR PROVIDER <PROVID"
ENDCASE
--*
CASE ADD1
MATCH PROVID
    ON NOMATCH INCLUDE
    ON MATCH TYPE ON PROVIDLG
        "PROVIDER <PROVID ON FILE ... CORRECT AND RESUBMIT
        "          LITHO # = <PRLTH"
        "-----*
    ON MATCH REJECT
ENDCASE
CASE CHANGE1
MATCH PROVID
    ON MATCH UPDATE PROVDATA PRLNME PRFNME PRINIT PRSTAT PRCATEG
    ON MATCH UPDATE PRPOS PRPAYGR PRJCODE PRSSN PRLTH
    ON NOMATCH TYPE ON PROVIDLG
        "PROVIDER <PROVID CHANGE REJECTED... CORRECT AND R

```

```

"          LITHO # = <PRLTH"
"-----
ON NOMATCH REJECT
ENDCASE
CASE DELETE1
MATCH PROVID
  ON MATCH TYPE ON PROVIDLG
    "PRVIDER <PROVID DELETED. FOLLOWING WAS FILE CONTENT
    " <D.PROVDATE <D.PRCATEG <D.PRLNME <D.PRFNME <D.PRI
    " <D.PRSSN <D.PRSTAT <D.PRPOS <D.PRPAYGR <PRJCODE"
    " LITHO = <D.PRLTH"
    "-----
  ON MATCH DELETE
  ON NOMATCH TYPE ON PROVIDLG
    "PROVIDER <PROVID DELETE REJECTED. CORRECT AND RES
    "          LITHO # = <PRLTH"
    "-----
  ON NOMATCH REJECT
ENDCASE
CHECK OFF
DATA ON RXXF300
END

```

```

--*-----*
--*
--*PROCEDURE RXXF400.FEX                PREPARED BY: D.R.BOLLING *
--*                                ON: 05/04/87 *
--*LOADS THE VISIT DATA BASE , ALL SEGMENTS *
--*-----*
--*
FILEDEF RXXF400 DISK D:VISIT.OUT
FILEDEF VISITLG DISK C:VISIT.LOG
-START
DOS STATE E:VISIT.FOC
-RUN
-IF &RETCODE NE 0 GOTO UNEW ;
USE E:VISIT.FOC
END
-GOTO CONT
-UNEW
USE E:VISIT.FOC NEW
END
CREATE FILE VISIT
-CONT
-CRTCLEAR
MODIFY FILE VISIT
--*
LOG DUPL MSG OFF
LOG NOMATCH MSG OFF
TYPE AT START ON VISITLG "ENCOUNTER LOAD PROCEDURE ERROR LOG
September 10, 1988 "
TYPE AT START "ENCOUNTER LOAD PROCEDURE IN PROGRESS  September 10, 198
CHECK OFF
COMPUTE
  RECTYPE/A1 =; VISIT_CNT/I1=0 ; TIME1/I3=0 ; PROV2/A5=' ' ;
  TIME2/I3=0 ; PROV2RES/A1=' ' ; APPSTAT/A1=' ' ; PLACE/A1=' ' ;
  REFERAL/A4=' ' ; JOBREL/A1=' ' ; DUTY/A1=' ' ; QTR/A1=' ' ;
  PROF/A1=' ' ; NAVAIL/A1=' ' ; ADMIT/A1=' ' ; ILLN/A1=' ' ;
  INJ/A1=' ' ; PURP/A1=' ' ; RULE/A1=' ' ; DX1CODE/A5=' ' ;
FIXFORM FNUM/A2 X1 RECTYPE/A1 CLINIC/C4 VDATE/C6 PROV1/C5 PTID/C11
FIXFORM LITHO/C8
--*-----*
--*                                RECORD VALIDATION *
--*-----*
IF FNUM GT '96' GOTO TOP;
IF FNUM LT '40' GOTO TOP;
IF RECTYPE GT 'A' GOTO TOP;
IF RECTYPE LT '1' GOTO TOP;
--*-----*
--*                                LOAD OF THE VISIT KEY DATA FOR ALL FORMS. *
--*-----*
MATCH CLINIC
  ON MATCH CONTINUE
  ON NOMATCH INCLUDE
  ON NOMATCH GOTO TESTRECT

```

```

MATCH VDATE
  ON MATCH CONTINUE
  ON NOMATCH INCLUDE
  ON NOMATCH GOTO TESTRECT
MATCH PROV1
  ON MATCH CONTINUE
  ON NOMATCH INCLUDE
  ON NOMATCH GOTO TESTRECT
MATCH PTID
  ON MATCH CONTINUE
  ON NOMATCH INCLUDE
  ON NOMATCH GOTO TESTRECT
MATCH LITHO
  ON MATCH GOTO TESTRECT
  ON NOMATCH INCLUDE
  ON NOMATCH GOTO TESTRECT
-*
```

```

CASE TESTRECT
  COMPUTE
    CLINIC = CLINIC;
    VDATE  = VDATE;
    PROV1  = PROV1;
    PTID   = PTID;
    LITHO  = LITHO;
    FNUM   = FNUM;
    IF RECTYPE EQ '1' GOTO PROC1TP;
    IF RECTYPE EQ '2' GOTO PROC2TP;
    IF RECTYPE EQ '3' GOTO PROC3TP;
    IF RECTYPE EQ '4' GOTO PROC4TP;
    IF RECTYPE EQ '5' GOTO PROC5TP;
    IF RECTYPE EQ '6' GOTO PROC6TP;
    IF RECTYPE EQ '7' GOTO PROC7TP;
    IF RECTYPE EQ '8' GOTO PROC8TP;
    IF RECTYPE EQ '9' GOTO PROC9TP;
    IF RECTYPE EQ 'A' GOTO PROCATP;
  ENDCASE

```

```

-*****
-*          RECORD TRANSACTION TYPE '1' PROCESSING BEGINS
-*****

```

```

CASE PROC1TP
  MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
  MATCH VDATE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
  MATCH PROV1
    ON MATCH CONTINUE
    ON NOMATCH REJECT
  MATCH PTID
    ON MATCH CONTINUE
    ON NOMATCH REJECT

```

-*

MATCH LITHO

ON MATCH

FIXFORM X-38 X40

FIXFORM VCNT/1 TIME1/3 PROV2/5 TIME2/3 PROV2RES/1

FIXFORM APPSTAT/1 REFERRAL/4 PLACE/1 JOBREL/1 DUTY/1

FIXFORM QTR/1 PROF/1 NAVAIL/1 ADMIT/1 ILLN/1

FIXFORM INJ/1 PURP/1 RULE/1 DX1CODE/5

ON MATCH UPDATE VCNT TIME1 PROV2 TIME2 PROV2RES APPSTAT PLACE

ON MATCH UPDATE REFERRAL JOBREL DUTY QTR PROF NAVAIL ADMIT ILLN

ON MATCH UPDATE INJ PURP RULE DX1CODE

ON NOMATCH REJECT

GOTO TOP

ENDCASE

-*

- * RECORD TRANSACTION TYPE '2' PROCESSING BEGINS

-*

CASE PROC2TP

MATCH CLINIC

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH VDATE

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH PROV1

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH PTID

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH LITHO

ON MATCH

FIXFORM X-38 X40 PCODE/1 PCODE/5

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH PCODE

ON MATCH REJECT

ON NOMATCH INCLUDE

GOTO TOP

ENDCASE

-*

- * RECORD TRANSACTION TYPE '3' PROCESSING BEGINS

-*

CASE PROC3TP

MATCH CLINIC

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH VDATE

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH PROV1

ON MATCH CONTINUE

ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-38 X40 SPCC/1
ON MATCH COMPUTE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH SPCC
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

-* RECORD TRANSACTION TYPE '4' PROCESSING BEGINS

CASE PROC4TP
MATCH CLINIC
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH VDATE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PROV1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-38 X40 OTHER/5
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH OTHER
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

-* RECORD TRANSACTION TYPE '5' PROCESSING BEGINS

CASE PROC5TP
MATCH CLINIC
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH VDATE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PROV1

```

ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-38 X40 UIC/6 TIMPR1/3 TIMPR2/3 TIMTR1/3 TIMTR2/3 MACT/3 OACT
FIXFORM RMIL/3 DEPD/3 CIVIL/3 CSHEET/1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH UIC
ON MATCH REJECT
ON NOMATCH TYPE " <UIC <CSHEET "
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

```

-----
-* RECORD TRANSACTION TYPE '6' PROCESSING BEGINS
-----

```

```

CASE PROC6TP
MATCH CLINIC
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH VDATE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PROV1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-38 X40 SPROG/1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH SPROG
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

```

-----
-* RECORD TRANSACTION TYPE '7' PROCESSING BEGINS
-----

```

```

CASE PROC7TP
MATCH CLINIC
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH VDATE
ON MATCH CONTINUE

```

ON NOMATCH REJECT
MATCH PROV1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-38 X40 DX2CODE/5
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH DX2CODE
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

-* RECORD TRANSACTION TYPE '8' PROCESSING BEGINS

CASE PROC8TP
MATCH CLINIC
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH VDATE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PROV1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-38 X40 ASSESMNT/1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH ASSESMNT
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

-* RECORD TRANSACTION TYPE '9' PROCESSING BEGINS

CASE PROC9TP
MATCH CLINIC
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH VDATE
ON MATCH CONTINUE

ON NOMATCH REJECT
MATCH PROV1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-38 X40 DISP/2
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH DISP
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

-*
-* RECORD TRANSACTION TYPE 'A' PROCESSING BEGINS
-*

CASE PROCATP
MATCH CLINIC
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH VDATE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PROV1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-38 X40 GPATID/11
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH GPATID
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

DATA ON RXXF400
END

```

-*****
-*   RXFCCL.FEX           D R BOLLING           03/18/86   *
-*   *
-*   CONTROL PROGRAM TO CORRECT CLINIC CODES IN VISIT FILE *
-*   *
-*****
-SET &&CLOLD='  ';
-SET &&CLNEW='  ';
-SET &&RTN='  ';
-SET &EMSG='  ';
-SCR
-CRTFORM LINE 3
-"      CORRECT CLINIC CODE IN VISIT FILE "
-" "
-"ENTER OLD CLINIC ID: <.IRHWT.&&CLOLD>"
-" "
-"ENTER NEW CLINIC ID: <.IRHWT.&&CLNEW>"
-" "
-" "
-"      X TO RETURN TO PREVIOUS MENU"
-"      E TO CHANGE OLD CLINIC CODE TO NEW CLINIC CODE"
-"      SELECT: <.IRHW.&&RTN"
-" "
-"HIT ENTER TO CONTINUE"
-" "
-" "
-" "
-"      <.YELL.D.&EMSG>"
-SET &EMSG='PLEASE ENTER DATA FOR ALL FIELDS.';
-IF &&RTN EQ 'X' THEN GOTO EXIT;
-IF &&CLOLD EQ '  ' THEN GOTO SCR;
-IF &&CLNEW EQ '  ' THEN GOTO SCR;
-IF &&RTN NE 'E' THEN GOTO SCR;
-*****
-* ENSURE NO CURRENT HOLD FILE
DOS STATE C:HOLD.FTM
-RUN
-IF &RETCODE NE 0 GOTO EXRXX;
DOS ERASE HOLD.FTM
-*****
-EXRXX
EX RXCCLD
-RUN
-*****
-* DID ANY RECORDS GET CREATED?
DOS STATE C:HOLD.FTM
-RUN
-IF &RETCODE EQ 0 GOTO NEXPROG;
-SET &EMSG='NO DATA WAS FOUND WITH THE ABOVE OLD CLINIC CODE.';
-GOTO SCR
-NEXPROG
EX RXCORL

```

-RUN
FILEDEF DELE DISK C:HOLD.FTM
-RUN
-WRITE DELE &&CLOLD
MODIFY FILE VISIT
FIXFORM CLINIC/4
 MATCH CLINIC
 ON MATCH DELETE
 ON NOMATCH REJECT
DATA ON DELE
END
-RUN
-EXIT

```

-*****
-*   RXXFCPA.FEX           D R BOLLING           03/18/86   *
-*                                     *
-*   CONTROL PROGRAM TO CORRECT PATIENT CODES IN VISIT FILE *
-*                                     *
-*****
-SET &SSNOLD='           ' ;
-SET &SSNNEW='           ' ;
-SET &FMPOLD='           ' ;
-SET &FMPNEW='           ' ;
-SET &&RTN=' ' ;
-SET &EMSG=' ' ;
--SCR
-CRTFORM LINE 3
-"      CORRECT PATIENT CODE IN VISIT FILE "
-" "
-"ENTER OLD PATIENT SSN: <.IRHWT.&SSNOLD>      FMP: <.IRHWT.&FMPOLD>"
-" "
-"ENTER NEW PATIENT SSN: <.IRHWT.&SSNNEW>      FMP: <.IRHWT.&FMPNEW>"
-" "
-" "
-"      X TO RETURN TO PREVIOUS MENU"
-"      E TO CHANGE OLD PATIENT CODE TO NEW PATIENT CODE"
-"      SELECT: <.IRHW.&&RTN"
-" "
-"HIT ENTER TO CONTINUE"
-" "
-" "
-" "
-"      <.YELL.D.&EMSG>"
-SET &EMSG='PLEASE ENTER DATA FOR ALL FIELDS.' ;
-IF &&RTN EQ 'X' THEN GOTO EXIT;
-IF &SSNOLD EQ ' ' THEN GOTO SCR;
-IF &SSNNEW EQ ' ' THEN GOTO SCR;
-SET &FO='1' | &FMPOLD;
-SET &FN='1' | &FMPNEW;
-IF &FO EQ '1' ' THEN GOTO SCR;
-IF &FN EQ '1' ' THEN GOTO SCR;
-IF &&RTN NE 'E' THEN GOTO SCR;
-SET &&PAOLD= &SSNOLD | &FMPOLD ;
-SET &&PANEW= &SSNNEW | &FMPNEW ;
-*****
-*   ENSURE NO CURRENT HOLD FILE      *
DOS STATE C:HOLD.FTM
-RUN
-IF &RETCODE NE 0 GOTO EXRXX;
DOS ERASE HOLD.FTM
-*****
-EXRXX
EX RXXCPAD
-RUN
-*****

```

```

-* DID ANY RECORDS GET CREATED?      *
DOS STATE C:HOLD.FTM
-RUN
-IF &RETCODE EQ 0 GOTO NEXPROG;
-SET &EMSG='NO DATA WAS FOUND WITH THE ABOVE OLD PATIENT ID.';
-GOTO SCR
-*****
-NEXPROG
EX RXXCORL
-RUN
FILEDEF HOLD DISK C:HOLD.FTM
-RUN
TABLE FILE VISIT.PATIENT_ID
PRINT CLINIC VISIT_DATE PROV_ID PATIENT_ID
IF PATIENT_ID EQ &&PAOLD
ON TABLE SAVE AS HOLD
END
-RUN
FILEDEF DELE DISK C:HOLD.FTM
-RUN
MODIFY FILE VISIT
FIXFORM CLINIC/4 VISIT_DATE/6 PROV_ID/5 PATIENT_ID/11
MATCH CLINIC
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH VISIT_DATE
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PROV_ID
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PATIENT_ID
  ON MATCH DELETE
  ON NOMATCH REJECT
DATA ON DELE
END
-RUN
-EXIT

```

```

-*****
-*   RXXFCPR.FEX           D R BOLLING           03/18/86   *
-*                                                                *
-*   CONTROL PROGRAM TO CORRECT PROVIDER CODES IN VISIT FILE *
-*                                                                *
-*****
-SET &&PROLD='      ';
-SET &&PRNEW='      ';
-SET &&RTN='  ';
-SET &EMSG='  ';
-SCR
-CRTFORM LINE 3
-"      CORRECT PROVIDER CODE IN VISIT FILE "
-" "
-"ENTER OLD PROVIDER ID: <.IRHWT.&&PROLD>"
-" "
-"ENTER NEW PROVIDER ID: <.IRHWT.&&PRNEW>"
-" "
-" "
-"      X TO RETURN TO PREVIOUS MENU"
-"      E TO CHANGE OLD PROVIDER CODE TO NEW PROVIDER CODE"
-"      SELECT: <.IRHW.&&RTN"
-" "
-"HIT ENTER TO CONTINUE"
-" "
-" "
-" "
-"      <.YELL.D.&EMSG>"
-SET &EMSG='PLEASE ENTER DATA FOR ALL FIELDS.';
-IF &&RTN EQ 'X' THEN GOTO EXIT;
-IF &&PROLD EQ '  ' THEN GOTO SCR;
-IF &&PRNEW EQ '  ' THEN GOTO SCR;
-IF &&RTN NE 'E' THEN GOTO SCR;
-*****
-* ENSURE NO CURRENT HOLD FILE      *
DOS STATE C:HOLD.FTM
-RUN
-IF &RETCODE NE 0 GOTO EXRXX;
DOS ERASE HOLD.FTM
-*****
-EXRXX
EX RXXCPRD
-RUN
-*****
-* DID ANY RECORDS GET CREATED?      *
DOS STATE C:HOLD.FTM
-RUN
-IF &RETCODE EQ 0 GOTO NEXPROG;
-SET &EMSG='NO DATA WAS FOUND WITH THE ABOVE OLD PROVIDER CODE.';
-GOTO SCR
-*****
-NEXPROG

```

```
EX RXXCORL
-RUN
FILEDEF HOLD DISK C:HOLD.FTM
-RUN
TABLE FILE VISIT.PROV_ID
PRINT CLINIC VISIT_DATE PROV_ID
IF PROV_ID EQ &&PROLD
ON TABLE SAVE AS HOLD
END
-RUN
FILEDEF DELE DISK C:HOLD.FTM
-RUN
MODIFY FILE VISIT
FIXFORM CLINIC/4 VISIT_DATE/6 PROV_ID/5
  MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
  MATCH VISIT_DATE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
  MATCH PROV_ID
    ON MATCH DELETE
    ON NOMATCH REJECT
DATA ON DELE
END
-RUN
-EXIT
```

```

*****
-*
-*      RXXFDCL.FEX                      DUMP CLINIC.FOC FOR SHIPMENT *
-*
-*                               3/4/86      D R BOLLING      *
-*      CHANGE DRIVE FOR SITE TO D      9/30/86      D R BOLLING      *
-*      CHANGE DRIVE FOR SITE TO C      12/16/86      D R BOLLING      *
-*      PUTTING SYSTEM BACK ON C
*****
-CRTCLEAR
-SET &SEL=' ';
-MENU
-CRTFORM
-"
-"
-"          CLINIC DATA
-"
-"      PREPARE BERNOULLI CARTRIDGE FOR SHIPMENT TO FT. DETRICK
-"
-"          1. PLACE NEWLY FORMATED CARTRIDGE IN DRIVE E.
-"
-"          2. DRIVE D SHOULD CONTAIN CLINIC.FOC FILE (STANDARD).
-"
-"      X ... RETURN TO PREVIOUS MENU      C ... CONTINUE
-"
-"
-"          ENTER YOUR SELECTION .....<&SEL  <70 |"
-"
-IF &SEL EQ X GOTO EXIT ELSE IF &SEL EQ C GOTO CONTI
-ELSE GOTO MENU;
-CONTI
-CRTCLEAR
-TYPE
-TYPE      PREPARATION OF FT. DETRICK BERNOULLI CLINIC FILE
-TYPE
-TYPE          IN PROGRESS
-TYPE
USE C:SITE.IDF
END
FILEDEF SITEF DISK C:SITE.IDF
-RUN
-SET &&SITE=' ';
-READ SITEF &&SITE
USE
      D:CLINIC.FOC
      E:DETCLIN.FTM NEW
END
FILEDEF DETR DISK E:DETCLIN.FTM
-SET ALL=ON;
DEFINE FILE CLINIC

```

```
SITE/A4='&&SITE';  
END  
-RUN  
-TYPE CLINIC RECORDS BEING PROCESSED  
TABLEF FILE CLINIC  
PRINT SITE SEG.CLINIC  
ON TABLE SAVE AS DETR  
END  
-RUN  
-EXIT
```



```

SITE/A4='&&SITE';
FMP/A2= EDIT(PTID,'$$$$$$$$$99');
C1/A1= EDIT(PTID,'9$$$$$$$$$');
C2/A1= EDIT(PTID,'$9$$$$$$$$$');
C3/A1= EDIT(PTID,'$$9$$$$$$$$$');
C4/A1= EDIT(PTID,'$$$9$$$$$$$$$');
C5/A1= EDIT(PTID,'$$$$9$$$$$$$$$');
C6/A1= EDIT(PTID,'$$$$$9$$$$$$$$$');
C7/A1= EDIT(PTID,'$$$$$$9$$$$$$$$$');
C8/A1= EDIT(PTID,'$$$$$$$9$$$$$');
C9/A1= EDIT(PTID,'$$$$$$$$$9$$$');
H1/A1=DECODE C1(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H2/A1=DECODE C2(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H3/A1=DECODE C3(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H4/A1=DECODE C4(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H5/A1=DECODE C5(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H6/A1=DECODE C6(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H7/A1=DECODE C7(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H8/A1=DECODE C8(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H9/A1=DECODE C9(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
NSSN/A9=H6|H8|H1|H5|H2|H9|H4|H7|H3;
C1/A1= EDIT(PT_DUAL_SSN,'9$$$$$$$$$');
C2/A1= EDIT(PT_DUAL_SSN,'$9$$$$$$$$$');
C3/A1= EDIT(PT_DUAL_SSN,'$$9$$$$$$$$$');
C4/A1= EDIT(PT_DUAL_SSN,'$$$9$$$$$$$$$');
C5/A1= EDIT(PT_DUAL_SSN,'$$$$9$$$$$$$$$');
C6/A1= EDIT(PT_DUAL_SSN,'$$$$$9$$$$$$$$$');
C7/A1= EDIT(PT_DUAL_SSN,'$$$$$$9$$$$$$$$$');
C8/A1= EDIT(PT_DUAL_SSN,'$$$$$$$9$$$$$');
C9/A1= EDIT(PT_DUAL_SSN,'$$$$$$$$$9$$$');
H1/A1=DECODE C1(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H2/A1=DECODE C2(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H3/A1=DECODE C3(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H4/A1=DECODE C4(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H5/A1=DECODE C5(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H6/A1=DECODE C6(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
9 '3' 0 '6');
H7/A1=DECODE C7(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'

```

```

      9 '3' 0 '6');
H8/A1=DECODE C8(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
      9 '3' 0 '6');
H9/A1=DECODE C9(1 '4' 2 '0' 3 '8' 4 '1' 5 '7' 6 '9' 7 '5' 8 '2'
      9 '3' 0 '6');
DSSN/A9=H6|H8|H1|H5|H2|H9|H4|H7|H3;
END
-RUN
-TYPE PATIENT RECORDS BEING PROCESSED
TABLEF FILE PATIENT
PRINT SITE NSSN FMP PT_DOB PT_CATEGORY PT_LITHO PT_REGDATE
      PT_GENDER PT_RACE PT_PREFIX PT_PAY_GRADE PT_JOB_CODE PT_LOCATION
      PT_TRN_TDY PT_VAELIG PT_HCI PT_ZIP_CODE DSSN PT_FOREIGN
ON TABLE SAVE AS DETR
END
-RUN
-EXIT

```

```

-*****
-*
-*   RXXFDPR.FEX                      DUMP PROVIDER.FOC FOR SHIPMENT
-*
-*                               3/4/86      D R BOLLING
-*   REMOVE NAME AND SSN          4/2/86      D R BOLLING
-*   CHANGE DRIVE FOR SITE TO D    9/30/86     D R BOLLING
-*   CHANGE DRIVE BACK TO D        12/16/86    D R BOLLING
-*****
-CRTCLEAR
-SET &SEL=' ';
-MENU
-CRTFORM
-"
-"
-"          PROVIDER DATA
-"
-"   PREPARE BERNOULLI CARTRIDGE FOR SHIPMENT TO FT. DETRICK
-"
-"       1. PLACE NEWLY FORMATED CARTRIDGE IN DRIVE E.
-"
-"       2. DRIVE D SHOULD CONTAIN PROVIDER.FOC FILE (STANDARD).
-"
-"   X ... RETURN TO PREVIOUS MENU      C ... CONTINUE
-"
-"
-"          ENTER YOUR SELECTION .....<&SEL  <70 |"
-"
-"
-IF &SEL EQ X GOTO EXIT ELSE IF &SEL EQ C GOTO CONTI
-ELSE GOTO MENU;
-CONTI
-CRTCLEAR
-TYPE
-TYPE   PREPARATION OF FT. DETRICK BERNOULLI PROVIDER FILE
-TYPE
-TYPE           IN PROGRESS
-TYPE
USE C:SITE.IDF
END
FILEDEF SITEF DISK C:SITE.IDF
-RUN
-SET &&SITE=' ';
-READ SITEF &&SITE
USE
    D:PROVIDER.FOC
    E:DETPROV.FTM NEW
END
FILEDEF DETR DISK E:DETPROV.FTM
-SET ALL=ON;
DEFINE FILE PROVIDER

```

```
SITE/A4='&&SITE';  
END  
-RUN  
-TYPE PROVIDER RECORDS BEING PROCESSED  
TABLEF FILE PROVIDER  
PRINT SITE PROV_ID PROV_DATE PROV_STAT PROV_CATEG PROV_PRPOS PROV_PAYG  
      PROV_JOBCODE PROV_LITHO  
ON TABLE SAVE AS DETR  
END  
-RUN  
-EXIT
```

```

*****
-*
-* RXXFDVI.FEX DUMP VISIT.FOC FOR SHIPMENT *
-* 5/4/87 D R BOLLING *
-*
-* THIS PROGRAM ENCRYPTS SSN AND DUMPS DATA TO D DRIVE *
*****

```

```

-RUN
-CRTCLEAR
-SET &SEL=' ';
-MENU
-CRTFORM

```

```

-"
-"
-" VISIT DATA
-"
-" PREPARE BERNOULLI CARTRIDGE FOR SHIPMENT TO FT. DETRICK
-"
-" 1. PLACE NEWLY FORMATED CARTRIDGE IN DRIVE D.
-"
-" 2. VISIT.FOC FILE IN DRIVE E WILL BE DUMPED TO D.
-"
-" X ... RETURN TO PREVIOUS MENU C ... CONTINUE
-"
-"
-"
-"
-"
-
```

```

-" ENTER YOUR SELECTION .....<&SEL <70 |"
-
```

```

-IF &SEL EQ X GOTO EXIT ELSE IF &SEL EQ C GOTO CONTI
-ELSE GOTO MENU;
-CONTI
-CRTCLEAR
-TYPE
-TYPE PREPARATION OF FT. DETRICK BERNOULLI
-TYPE
-TYPE IN PROGRESS
-TYPE
USE C:SITE.IDF
END
FILEDEF SITEF DISK C:SITE.IDF
-RUN
-SET &&SITE=' ';
-READ SITEF &&SITE
USE
E:VISIT.FOC
D:DETRICK.FTM NEW
END
FILEDEF DETR DISK D:DETRICK.FTM
-SET ALL=ON;
-SET &&RECTYPE='1';
DEFINE FILE VISIT

```

```

SITE/A4='&&SITE';
RECTP/A1='&&RECTYPE';
FMP/A2= EDIT(PTID,'$$$$$$$$$99');
SSNUM/A9=EDIT(PATIENT_ID,'999999999$$');
NUMT/I9=EDIT(SSNUM);
K1/I2=NUMT/33554432;
NUMT/I9=NUMT-K1*33554432;
K2/I2=NUMT/1048576;
NUMT/I9=NUMT-K2*1048576;
K3/I2=NUMT/32768;
NUMT/I9=NUMT-K3*32768;
K4/I2=NUMT/1024;
NUMT/I9=NUMT-K4*1024;
K5/I2=NUMT/32;
NUMT/I9=NUMT-K5*32;
K6/I2=NUMT;
KD1/A1=DECODE K1(0 'A' 1 'B' 2 'C' 3 'D' 4 'E' 5 'F' 6 'G' 7 'H' 8 'I'
9 'J' 10 'K' 11 'L' 12 'M' 13 'N' 14 'O' 15 'P' 16 'Q' 17 'R'
18 'S' 19 'T' 20 'U' 21 'V' 22 '2' 23 '3' 24 '4' 25 '5' 26 '6'
27 '7' 28 '8' 29 '9' 30 '0' 31 '1');
KD2/A1=DECODE K2(0 'A' 1 'B' 2 'C' 3 'D' 4 'E' 5 'F' 6 'G' 7 'H' 8 'I'
9 'J' 10 'K' 11 'L' 12 'M' 13 'N' 14 'O' 15 'P' 16 'Q' 17 'R'
18 'S' 19 'T' 20 'U' 21 'V' 22 '2' 23 '3' 24 '4' 25 '5' 26 '6'
27 '7' 28 '8' 29 '9' 30 '0' 31 '1');
KD3/A1=DECODE K3(0 'A' 1 'B' 2 'C' 3 'D' 4 'E' 5 'F' 6 'G' 7 'H' 8 'I'
9 'J' 10 'K' 11 'L' 12 'M' 13 'N' 14 'O' 15 'P' 16 'Q' 17 'R'
18 'S' 19 'T' 20 'U' 21 'V' 22 '2' 23 '3' 24 '4' 25 '5' 26 '6'
27 '7' 28 '8' 29 '9' 30 '0' 31 '1');
KD4/A1=DECODE K4(0 'A' 1 'B' 2 'C' 3 'D' 4 'E' 5 'F' 6 'G' 7 'H' 8 'I'
9 'J' 10 'K' 11 'L' 12 'M' 13 'N' 14 'O' 15 'P' 16 'Q' 17 'R'
18 'S' 19 'T' 20 'U' 21 'V' 22 '2' 23 '3' 24 '4' 25 '5' 26 '6'
27 '7' 28 '8' 29 '9' 30 '0' 31 '1');
KD5/A1=DECODE K5(0 'A' 1 'B' 2 'C' 3 'D' 4 'E' 5 'F' 6 'G' 7 'H' 8 'I'
9 'J' 10 'K' 11 'L' 12 'M' 13 'N' 14 'O' 15 'P' 16 'Q' 17 'R'
18 'S' 19 'T' 20 'U' 21 'V' 22 '2' 23 '3' 24 '4' 25 '5' 26 '6'
27 '7' 28 '8' 29 '9' 30 '0' 31 '1');
KD6/A1=DECODE K6(0 'A' 1 'B' 2 'C' 3 'D' 4 'E' 5 'F' 6 'G' 7 'H' 8 'I'
9 'J' 10 'K' 11 'L' 12 'M' 13 'N' 14 'O' 15 'P' 16 'Q' 17 'R'
18 'S' 19 'T' 20 'U' 21 'V' 22 '2' 23 '3' 24 '4' 25 '5' 26 '6'
27 '7' 28 '8' 29 '9' 30 '0' 31 '1');
NSSN/A6=KD1|KD2|KD3|KD4|KD5|KD6;
END
-RUN
- TYPE RECORDS TYPE 1 BEING PROCESSED
TABLEF FILE VISIT.TIME1
PRINT SITE RECTP CLINIC VDATE PROV1 NSSN FMP SEG.LITHO
ON TABLE SAVE AS DETR
END
-RUN
FILEDEF DETR DISK D:DETRICK.FTM APPEND
-SET &&RECTYPE='2';
DEFINE FILE VISIT ADD

```

```

    RECTP/A1='&&RECTYPE';
END
-RUN
- TYPE RECORDS TYPE 2 BEING PROCESSED
TABLEF FILE VISIT.PCODE
PRINT SITE RECTP CLINIC VDATE PROV1 NSSN FMP LITHO FNUM SEG.PCODE
ON TABLE SAVE AS DETR
END
-RUN
-SET &&RECTYPE='3';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
END
-RUN
- TYPE RECORDS TYPE 3 BEING PROCESSED
TABLEF FILE VISIT.SPCC
PRINT SITE RECTP CLINIC VDATE PROV1 NSSN FMP LITHO FNUM SPCC
ON TABLE SAVE AS DETR
END
-RUN
-SET &&RECTYPE='4';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
END
-RUN
- TYPE RECORDS TYPE 4 BEING PROCESSED
TABLEF FILE VISIT.OTHER
PRINT SITE RECTP CLINIC VDATE PROV1 NSSN FMP LITHO FNUM OTHER
ON TABLE SAVE AS DETR
END
-RUN
-SET &&RECTYPE='5';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
END
-RUN
- TYPE RECORDS TYPE 5 BEING PROCESSED
TABLEF FILE VISIT.UIC
PRINT SITE RECTP CLINIC VDATE PROV1 NSSN FMP LITHO FNUM SEG.UIC
ON TABLE SAVE AS DETR
END
-RUN
-SET &&RECTYPE='6';
DEFINE FILE VISIT ADD
    RECTP/A1='&&RECTYPE';
END
-RUN
- TYPE RECORDS TYPE 6 BEING PROCESSED
TABLEF FILE VISIT.SPROG
PRINT SITE RECTP CLINIC VDATE PROV1 NSSN FMP LITHO FNUM SPROG
ON TABLE SAVE AS DETR
END

```


-"
-"
-"
-"
-"
-"
-"
-"
-"
-"

DUMP IS COMPLETE

HIT ENTER TO CONTINUE

-RUN
-EXIT

VISIT.MAS

FILE=VISIT,SUFFIX=FOC,\$
SEGNAME=VISLOC,SEGTYPE=S1,\$
FIELD=CLINIC,CL_UCA,A4,\$
SEGNAME=VISDTE,PARENT=VISLOC,SEGTYPE=S1,\$
FIELD=VISIT_DATE,VDATE,I6YMD,\$
SEGNAME=VISPROV,PARENT=VISDTE,SEGTYPE=S1,\$
FIELD=PROV_ID,PROV1,A5,\$
SEGNAME=VISITOR,PARENT=VISPROV,SEGTYPE=S1,\$
FIELD=PATIENT_ID,PTID,A11,\$
SEGNAME=LITHO,PARENT=VISITOR,SEGTYPE=S1,\$
FIELD=LITHO_CODE,LITHO,I8,\$
FIELD=FORM_NUM,FNUM,A2,\$
FIELD=VISIT_CNT,VCNT,I1,\$
FIELD=PROV1_TIME,TIME1,I3,\$
FIELD=PROVIDER_2,PROV2,A5,\$
FIELD=PROV2_TIME,TIME2,I3,\$
FIELD=PROV2_RESON,PROV2RES,A1,\$
FIELD=APPT_STATUS,APPSTAT,A1,\$
FIELD=PLACE_OF_VIS,PLACE,A1,\$
FIELD=INP_REFERAL,REFERAL,A4,\$
FIELD=JOB_RELATED,JOBREL,A1,\$
FIELD=MIL_DUTY,DUTY,A1,\$
FIELD=MIL_QTR,QTR,A1,\$
FIELD=MIL_PROF,PROF,A1,\$
FIELD=NOT_AVAIL,NAVAIL,A1,\$
FIELD=ADMITTED,ADMIT,A1,\$
FIELD=INFELD,ILLN,A1,\$
FIELD=INJURIES,INJ,A1,\$
FIELD=PURP_VIS,PURP,A1,\$
FIELD=RO_F_UP,RULE,A1,\$
FIELD=DIAG1_CODE,DX1CODE,A5,\$
SEGNAME=V_PROC,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=PROC_CODE,PCODE,A5,\$
FIELD=PROV_CD,PCODE,A1,\$
SEGNAME=V_SPCC,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=SP_PRE_CODE,SPCC,A1,\$
SEGNAME=V_OTHER,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=OTHER_CODE,OTHER,A5,\$
SEGNAME=V_GROUP,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=UNIT_ID,UIC,A6,\$
FIELD=TIME_PR1,TIMPR1,I3,\$
FIELD=TIME_PR2,TIMPR2,I3,\$
FIELD=TIME_TR1,TIMTR1,I3,\$
FIELD=TIME_TR2,TIMTR2,I3,\$
FIELD=NO_ACT_DUTY,MACT,I3,\$
FIELD=NO_OTH_DUTY,OACT,I3,\$
FIELD=NO_RET_MIL,RMIL,I3,\$
FIELD=NO_DEPEND,DEPD,I3,\$
FIELD=NO_CIVIL,CIVIL,I3,\$
FIELD=CONT_SHEET,CSHEET,A1,\$

SEGNAME=V_SPEC,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=SPEC_PROG,SPROG,A1,\$
SEGNAME=V_DIAG2,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=DIAG2_CODE,DX2CODE,A5,\$
SEGNAME=V_PSY,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=ASSESMNT,ASSESMNT,A1,\$
SEGNAME=V_DISPOS,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=DISPOS,DISP,A2,\$
SEGNAME=V_GRSSN,PARENT=LITHO,SEGTYPE=S1,\$
FIELD=GR_PAT_ID,GPATID,A11,\$
END
DBA=LUGNUT,\$
USER=NUTMEG,ACCESS=RW,\$

APPENDIX E

MAINFRAME FOCUS LOAD PROGRAMS (REFERENCED SECTION 6.2.3)

APPENDIX E

MAINFRAME FOCUS LOAD PROGRAMS

TABLE OF CONTENTS

CLINLOAD	Clinic Description
DIAGLOAD	Diagnosis Description
NLOAD	New Design Visit Data
OTHLOAD	Other Code Description
PASILO	Patient Sidpers
PATILOAD	Patient
PATILD	Patient Deers
PROCLOAD	Procedure Description
PROVLOAD	Provider
RXXF400	Old Design Visit Data

```

--*-----
--*
--* PROCEDURE CLINLOAD.FEX          PREPARED BY D.R.BOLLING
--*                                ON 01/10/86
--* LOADS THE CLINIC FLAT FILE TO FOCUS FILE
--*          (DETCLIN.DATA)        (CLINIC.FOCUS)
--*
--*-----
TSO ALLOC F(CLINF) DA('DXB.POLK.CLINIC')
-CRTCLEAR
MODIFY FILE CLINIC
TYPE AT START "CLINIC LOAD PROCEDURE IN PROGRESS"
FIXFORM CLINIC/8   CL_TITLE/20  CL_302CODE/2
FIXFORM CL_302LINE/I3
--*
MATCH CLINIC
      ON NOMATCH INCLUDE
      ON   MATCH CONTINUE
CHECK OFF
DATA ON CLINF
END
TSO FREE F(CLINF)

```

```

--*-----
--*
--* PROCEDURE DIAGLOAD.FEX          PREPARED BY D.R.BOLLING
--*                                ON 01/10/86
--* LOADS THE DIAGNOSIS FLAT FILE TO FOCUS FILE
--*          (DETDIAG.DATA)        (DIAGNOS.FOCUS)
--*
--*-----
TSO ALLOC F(DIAGF) DA(DIAG.DEC87.CODES) SHR
-CRTCLEAR
MODIFY FILE NDIAG
TYPE AT START "DIAGNOSIS LOAD PROCEDURE IN PROGRESS"
FIXFORM DGNCD/5 X2 DGNDSC/50
--*
MATCH DIAG_CODE
      ON NOMATCH INCLUDE
      ON   MATCH UPDATE DGNDSC
CHECK OFF
DATA ON DIAGF
END

```

```

--*-----*
--*
--*PROCEDURE NLOAD.                                PREPARED BY: D.R.B.
--*                                                ON: 06/03/87
--*
--*LOADS THE REDESIGNED VISIT DATA BASE INTO MAINFRAME FOCUS
--*
--*
--*
SET BINS=64
TSO ALLOC F(DETRICK) DA('DXB.U.REDS.JUL87.VISIT')
-START
-RUN
-CRTCLEAR
MODIFY FILE NCAMP
--*
TYPE AT START "ENCOUNTER LOAD PROCEDURE IN PROGRESS
September 10, 1988"
CHECK OFF
COMPUTE
  RECTYPE/A1 =;
  VISIT CNT/I1=0; TIME1/I3=0; PROV2/A5=' ' ; TIME2/I3=0;
  PROV2RES/A1=' ' ; APPSTAT/A1=' ' ; REFERRAL/A4=' ' ;
  PLACE/A1=' ' ; JOBREL/A1=' ' ; DUTY/A1=' ' ; QTR/A1=' ' ;
  PROF/A1=' ' ; NAVAIL/A1=' ' ; ADMIT/A1=' ' ; ILLN/A1=' ' ;
  INJ/A1=' ' ; PURP/A1=' ' ; RULE/A1=' ' ; DX1CODE/A5=' ' ;
FIXFORM SITE/A4 RECTYPE/A1 CLINIC/C4 VDATE/C6 PROV1/C5
FIXFORM PTID/C8 LITHO/C8
--*
--*-----*
--*
--*                                RECORD VALIDATION
--*
--*
IF RECTYPE GT '9'    GOTO TOP;
IF RECTYPE LT 'A'    GOTO TOP;
--*
--*-----*
--*
--*                                LOAD OF THE VISIT KEY DATA FOR ALL FORMS.
--*
--*
MATCH SITE
  ON MATCH CONTINUE
  ON NOMATCH INCLUDE
  ON NOMATCH GOTO TESTRECT
MATCH CLINIC
  ON MATCH CONTINUE
  ON NOMATCH INCLUDE
  ON NOMATCH GOTO TESTRECT
MATCH VDATE
  ON MATCH CONTINUE
  ON NOMATCH INCLUDE
  ON NOMATCH GOTO TESTRECT
MATCH PROV1
  ON MATCH CONTINUE

```

```

    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT
MATCH PTID
    ON MATCH CONTINUE
    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT
MATCH LITHO
    ON MATCH GOTO TESTRECT
    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT

```

-*

CASE TESTRECT

COMPUTE

SITE = SITE;

CLINIC = CLINIC;

VDATE = VDATE;

PROV1 = PROV1;

PTID = PTID;

LITHO = LITHO;

IF RECTYPE EQ '1' GOTO VALIDFM;

IF RECTYPE EQ '2' GOTO PROC2TP;

IF RECTYPE EQ '3' GOTO PROC3TP;

IF RECTYPE EQ '4' GOTO PROC4TP;

IF RECTYPE EQ '5' GOTO PROC5TP;

IF RECTYPE EQ '6' GOTO PROC6TP;

IF RECTYPE EQ '7' GOTO PROC7TP;

IF RECTYPE EQ '8' GOTO PROC8TP;

IF RECTYPE EQ '9' GOTO PROC9TP;

IF RECTYPE EQ 'A' GOTO PROCATP;

ENDCASE

-*

CASE VALIDFM

-*

-*

-*

RECORD TRANSACTION TYPE '1' PROCESSING BEGINS

-*

-*

MATCH SITE

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH CLINIC

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH VDATE

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH PROV1

ON MATCH CONTINUE

ON NOMATCH REJECT

MATCH PTID

ON MATCH CONTINUE

ON NOMATCH REJECT

```

--*
MATCH LITHO
ON MATCH
FIXFORM X-36 X36
FIXFORM FNUM/2
FIXFORM VCNT/1      TIME1/3      PROV2/5      TIME2/3      PROV2RES/1
FIXFORM APPSTAT/1   PLACE/1      REFERAL/4    JOBREL/1     DUTY/1
FIXFORM QTR/1       PROF/1       NAVAIL/1     ADMIT/1      ILLN/1
FIXFORM INJ/1       PURP/1       RULE/1       DX1CODE/5
    ON MATCH UPDATE FNUM VCNT TIME1 PROV2 TIME2 PROV2RES
    ON MATCH UPDATE APPSTAT PLACE REFERAL JOBREL DUTY QTR
    ON MATCH UPDATE PROF NAVAIL ADMIT ILLN INJ PURP RULE
    ON MATCH UPDATE DX1CODE
    ON NOMATCH REJECT
GOTO TOP
ENDCASE
--*
CASE PROC2TP
MATCH SITE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH VDATE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PROV1
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PTID
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36 X38 PCODE/5 PRCODE/1
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PCODE
    ON MATCH REJECT
    ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-*****
CASE PROC3TP
MATCH SITE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH VDATE

```

ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PROV1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36 X38 SPCC/1
ON MATCH COMPUTE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH SPCC
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

CASE PROC4TP

MATCH SITE

ON MATCH CONTINUE
ON NOMATCH REJECT

MATCH CLINIC

ON MATCH CONTINUE
ON NOMATCH REJECT

MATCH VDATE

ON MATCH CONTINUE
ON NOMATCH REJECT

MATCH PROV1

ON MATCH CONTINUE
ON NOMATCH REJECT

MATCH PTID

ON MATCH CONTINUE
ON NOMATCH REJECT

MATCH LITHO

ON MATCH

FIXFORM X-36 X38 OTHER/5

ON MATCH CONTINUE
ON NOMATCH REJECT

MATCH OTHER

ON MATCH REJECT
ON NOMATCH INCLUDE

GOTO TOP

ENDCASE

CASE PROC5TP

MATCH SITE

ON MATCH CONTINUE
ON NOMATCH REJECT

MATCH CLINIC

```

      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH VDATE
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PROV1
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PTID
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36 X38 UIC/6 TIMPR1/3 TIMPR2/3 TIMTR1/3 TIMTR2/3
FIXFORM MACT/3 OACT/3 RMIL/3 DEPD/3 CIVIL/3 CSHEET/1
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH UIC
      ON      MATCH REJECT
      ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-*****
CASE PROC6TP
MATCH SITE
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH CLINIC
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH VDATE
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PROV1
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PTID
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36 X38 SPROG/1
      ON      MATCH CONTINUE
      ON NOMATCH REJECT
MATCH SPROG
      ON      MATCH REJECT
      ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-*****
CASE PROC7TP
MATCH SITE

```

```

      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH CLINIC
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH VDATE
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH PROV1
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH PTID
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36 X38 DX2CODE/5
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH DX2CODE
      ON   MATCH REJECT
      ON  NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

```

CASE PROC8TP
MATCH SITE
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH CLINIC
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH VDATE
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH PROV1
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH PTID
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36 X38 ASSESMNT/1
      ON   MATCH CONTINUE
      ON  NOMATCH REJECT
MATCH ASSESMNT
      ON   MATCH REJECT
      ON  NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

```

CASE PROC9TP
MATCH SITE
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH CLINIC
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH VDATE
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PROV1
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PTID
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36 X38 DISP/2
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH DISP
  ON MATCH REJECT
  ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

```

-*****

```

CASE PROCATP
MATCH SITE
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH CLINIC
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH VDATE
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PROV1
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PTID
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36 X38 GPATID/11
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH GPATID
  ON MATCH REJECT
  ON NOMATCH INCLUDE
GOTO TOP

```

ENDCASE

-*****

DATA ON DETRICK

END

TSO FREE F(DETRICK)

```

-----
-*
-*
-* PROCEDURE OTHLOAD.FEX                PREPARED BY D.R.BOLLING
-*                                     ON 05/29/87
-* LOADS THE OTHER FLAT FILE TO FOCUS FILE
-*                                     (DETPROC.DATA)      (PROCEDUR.FOCUS)
-*
-----
TSO ALLOC F(OTHCF) DA('DXB.OTHER.MAY87.CODE')
-CRTCLEAR
MODIFY FILE OTHER
TYPE AT START "OTHER LOAD PROCEDURE IN PROGRESS"
FIXFORM OTHER_CODE/5 X2 OTHER_DESCR/67
-*
MATCH OTHER_CODE
      ON NOMATCH INCLUDE
      ON MATCH CONTINUE
CHECK OFF
DATA ON OTHCF
END

```

```

-----
-*
-*
-* PROCEDURE PASILO .FEX          PREPARED BY D.R.BOLLING
-*                                ON 10/09/86
-* LOADS THE PATIENT FLAT FILE TO FOCUS FILE
-*                                (DETPATI.DATA)  (PATIENT.FOCUS)
-* REJECTS DUPLICATE TRANSACTIONS      FOR SIDPERS DATA
-----
SET BINS=64
TSO ALLOC F(PATIF) DA('DXB.POLK.AUG87.PATI')
-CRTCLEAR
MODIFY FILE PATIENT
TYPE AT START "PATIENT LOAD PROCEDURE IN PROGRESS"
FIXFORM PTID/12
FIXFORM PT_DOB/C6 CATEG/A3 PTLITHO/C8
FIXFORM REGDATE/C6 SEX/C1 RACE/C1 PREFIX/C2 PAYGRADE/C2
FIXFORM JOBCODE/C4 PTLOC/C6 TRAINEE/C1 VA/C1 HCI/C1
ZIP/C5
FIXFORM PTDUALSSN/9 PTFORGN/C3
MATCH PTID
      ON NOMATCH INCLUDE
      ON MATCH REJECT
-*
CHECK OFF
DATA ON PATIF
END
TSO FREE F(PATIF)

```

```

-----
--
-- PROCEDURE PATILOAD.FEX          PREPARED BY D.R.BOLLING
--                                ON 01/10/86
-- LOADS THE PATIENT FLAT FILE TO FOCUS FILE
--                                (DETPATI.DATA)  (PATIENT.FOCUS)
--
-----
SET BINS=63
TSO ALLOC F(PATIF) DA('DXB.REDS.NOV86.DPATI')
-CRTCLEAR
MODIFY FILE PATIENT
TYPE AT START "PATIENT LOAD PROCEDURE IN PROGRESS"
FIXFORM PTID/12
FIXFORM PT DOB/C6 CATEG/A3 PTLITHO/C8
FIXFORM REGDATE/C6 SEX/C1 RACE/C1 PREFIX/C2 PAYGRADE/C2
FIXFORM JOBCODE/C4 PTLOC/C6 TRAINEE/C1 VA/C1 HCI/C1 ZIP/C5
FIXFORM PTDUALSSN/9 PTFORGN/C3
MATCH PTID
    ON NOMATCH INCLUDE
    ON MATCH GOTO SEG2
--
CASE SEG2
MATCH PTLITHO
    ON NOMATCH REJECT
    ON MATCH UPDATE CATEG REGDATE SEX RACE PREFIX PAYGRADE
    ON MATCH UPDATE JOBCODE PTLOC TRAINEE VA HCI ZIP
    ON MATCH UPDATE PTDUALSSN PTFORG
ENDCASE
CHECK OFF
DATA ON PATIF
END
TSO FREE F(PATIF)

```

```

-----*
-*
-* PROCEDURE    PATILD.FEX                PREPARED BY D.R.BOLLING *
-*                                           ON 07/07/87          *
-* LOADS THE PATIENT FLAT FILE TO FOCUS FILE *
-*           (DETPATI.DATA)      (PATIENT.FOCUS) *
-*   LOADS DEERS DATA WITH UPDATE CAPABILITY *
-----*
SET BINS=63
TSO ALLOC F(PATIF) DA('DXB.REDS.MAY87.DPATI')
-CRTCLEAR
MODIFY FILE PATIENT
TYPE AT START "PATIENT LOAD PROCEDURE IN PROGRESS"
FIXFORM PTID/12 PT_DOB/C6 CATEG/C3 PTLITHO/C8 REGDATE/C6
FIXFORM SEX/C1 RACE/C1 PREFIX/C2 PAYGRADE/C2 JOBCODE/C4
FIXFORM PTLOC/C6 TRAINEE/C1 VA/C1 HCI/C1 ZIP/C5
FIXFORM PTDUALSSN/C9 PTFORGN/C3
MATCH PTID
    ON NOMATCH INCLUDE
    ON MATCH UPDATE CATEG REGDATE SEX RACE PREFIX PAYGRADE
    ON MATCH UPDATE JOBCODE PTLOC TRAINEE VA HCI ZIP
    ON MATCH UPDATE PTDUALSSN PTFORGN
CHECK OFF
DATA ON PATIF
END
TSO FREE F(PATIF)

```

```

-----
-*
-*
-* PROCEDURE PROCLOAD.FEX          PREPARED BY D.R.BOLLING
-*                                ON 01/10/86
-* LOADS THE PROCEDURE FLAT FILE TO FOCUS FILE
-*                                (DETPROC.DATA)    (PROCEDUR.FOCUS)
-*
-----
TSO ALLOC F(PROCF) DA(PROC.DEC87.CODES)
-CRTCLEAR
MODIFY FILE NPROC
TYPE AT START "PROCEDURE LOAD PROCEDURE IN PROGRESS"
FIXFORM PROC_CODE/5 X2 PROC_DESCR/67
-*
MATCH PROC_CODE
    ON NOMATCH INCLUDE
    ON MATCH UPDATE PROC_DESCR
CHECK OFF
DATA ON PROCF
END
TSO FREE F(PROCF)

```

```

-----
-*
-* PROCEDURE PROVLOAD.FEX                PREPARED BY D.R.BOLLING
-*                                     ON 01/10/86
-* LOADS THE PROVIDER FLAT FILE TO FOCUS FILE
-*          (DETPROV.DATA)          (PROVIDER.FOCUS)
-*
-----
TSO ALLOC F(PROVF) DA('DXB.PROV.FIXED')
-CRTCLEAR
MODIFY FILE PROVIDER
TYPE AT START "PROVIDER LOAD PROCEDURE IN PROGRESS"
FIXFORM PROVID/9    PROVDATA/6
FIXFORM PRSTAT/1    PRCATEG/2    PRPOS/2
FIXFORM PRPAYGR/2    PRJCODE/7    PRLTH/8
-*
MATCH PROVID
      ON NOMATCH INCLUDE
      ON    MATCH REJECT
CHECK OFF
DATA ON PROV F
END
TSO FREE F(PROVF)

```

```

--*-----
--*
--*PROCEDURE RXXF400.FEX                                PREPARED BY: D.R.B.
--*                                                    ON: 01/03/86
--*LOADS THE VISIT DATA BASE , ALL SEGMENTS INTO MAIN FRAME
--*
--* READS BASE 32 SSN
--* PREVENT DATA OFFSET BY MOVING FIXFORM AFTER LITHO MATCH
--*      05/20/86
--* ALLOW FOR FORM NUMBER FIELD                                07/01/86
--* LOADS DATA INTO      BAMC DATA BASE (VBAMC)            01/14/87
--*-----
--*
SET BINS=64
TSO ALLOC F(DETRICK) DA('DXB.U.BAMC.OLD87.VISIT') SHR
-START
-RUN
-CRTCLEAR
MODIFY FILE VBAMC
--*
TYPE AT START "ENCOUNTER LOAD PROCEDURE IN PROGRESS
September 10, 1988"
CHECK OFF
COMPUTE
  RECTYPE/A1 =;
--*VISRES/A1=' '; TIME1/I3=0; PROV2/A5=' '; TIME2/I3=0;
--*PROV2RES/A1=' '; PROV1SEEN/A1=' '; PROV1SAME/A1=' ';
--*PROV2SEEN/A1=' '; PROV2SAME/A1=' '; LAST12/A1=' ';
--*NEWPROB/A1=' '; APPSTAT/A1=' '; REFERRAL/A4=' ';
--*PLACE/A1=' '; JOBREL/A1=' '; DISPO/A1=' '; LAB/A1=' ';
--*RX/A1=' '; DX1RULE/A1=' '; DX1CODE/A5=' ';
FIXFORM SITE/A4
FIXFORM RECTYPE/A1
FIXFORM CLINIC/C4      VDATE/C6      PROV1/C5
FIXFORM PTID/C8      LITHO/C8
--*
--*-----
--*
--*                                RECORD VALIDATION
--*
IF RECTYPE GT '9'      GOTO TOP;
IF RECTYPE LT '1'      GOTO TOP;
--*
--*-----
--*
--*                                LOAD OF THE VISIT KEY DATA FOR ALL FORMS.
--*
MATCH SITE
  ON MATCH CONTINUE
  ON NOMATCH INCLUDE
  ON NOMATCH GOTO TESTRECT
MATCH CLINIC
  ON MATCH CONTINUE
  ON NOMATCH INCLUDE

```

```

    ON NOMATCH GOTO TESTRECT
MATCH VDATE
    ON MATCH CONTINUE
    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT
MATCH PROV1
    ON MATCH CONTINUE
    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT
MATCH PTID
    ON MATCH CONTINUE
    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT
MATCH LITHO
    ON MATCH GOTO TESTRECT
    ON NOMATCH INCLUDE
    ON NOMATCH GOTO TESTRECT

```

```

- *
CASE TESTRECT
COMPUTE
SITE = SITE;
CLINIC = CLINIC;
VDATE = VDATE;
PROV1 = PROV1;
PTID = PTID;
LITHO = LITHO;
IF RECTYPE EQ '1' GOTO VALIDFM;
IF RECTYPE EQ '2' GOTO PROC2TP;
IF RECTYPE EQ '3' GOTO PROC3TP;
IF RECTYPE EQ '4' GOTO PROC4TP;
IF RECTYPE EQ '5' GOTO PROC5TP;
IF RECTYPE EQ '6' GOTO PROC6TP;
IF RECTYPE EQ '7' GOTO PROC7TP;
IF RECTYPE EQ '8' GOTO PROC8TP;
IF RECTYPE EQ '9' GOTO PROC9TP;
ENDCASE

```

```

- * -----
CASE VALIDFM
- * -----

```

```

- *
- *             COMMON FORM
- *   RECORD TRANSACTION TYPE '1' PROCESSING BEGINS
- *
- *

```

```

MATCH SITE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH VDATE
    ON MATCH CONTINUE

```

```

ON NOMATCH REJECT
MATCH PROV1
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PTID
  ON MATCH CONTINUE
  ON NOMATCH REJECT
-*
MATCH LITHO
ON MATCH
FIXFORM X-36
FIXFORM X36 FNUM/C2
FIXFORM VISRES/C1 TIME1/C3 PROV2/C5 TIME2/C3 PROV2RES/C1
FIXFORM PROV1SEEN/C1 PROV1SAME/C1 PROV2SEEN/C1
FIXFORM PROV2SAME/C1 LAST12/C1 NEWPROB/C1
FIXFORM APPSTAT/C1 REFERRAL/C4 PLACE/C1 JOBREL/C1
FIXFORM DISPO/C1 LAB/C1 RX/C1
FIXFORM PRPVST/C1 RELSURV/C1 ADMREAS/C1
FIXFORM DX1RULE/C1 DX1CODE/C5
  ON NOMATCH REJECT
  ON MATCH CONTINUE TO VISRES
  ON NOMATCH INCLUDE
  ON MATCH REJECT
ENDCASE
-*
CASE PROC2TP
MATCH SITE
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH CLINIC
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH VDATE
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PROV1
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PTID
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36
FIXFORM X38
FIXFORM PCODE/5
  ON MATCH CONTINUE
  ON NOMATCH REJECT
MATCH PCODE
  ON MATCH REJECT
  ON NOMATCH INCLUDE
GOTO TOP

```

```

ENDCASE
-*****
CASE PROC3TP
MATCH SITE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH VDATE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PROV1
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PTID
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36
FIXFORM X38
FIXFORM SUPPDISP/1
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH SUPPDISP
    ON MATCH REJECT
    ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-*****
CASE PROC4TP
MATCH SITE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH VDATE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PROV1
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PTID
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36
FIXFORM X38
FIXFORM XRAY/1

```

ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH XRAY
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

CASE PROC5TP
MATCH SITE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH CLINIC
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH VDATE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PROV1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36
FIXFORM X38
FIXFORM OTHORD
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH OTHORD
ON MATCH REJECT
ON NOMATCH INCLUDE
GOTO TOP
ENDCASE

CASE PROC6TP
MATCH SITE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH CLINIC
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH VDATE
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PROV1
ON MATCH CONTINUE
ON NOMATCH REJECT
MATCH PTID
ON MATCH CONTINUE

```

    ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36
FIXFORM X38
FIXFORM SPROG/2
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH SPROG
    ON MATCH REJECT
    ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-*****
CASE PROC7TP
MATCH SITE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH VDATE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PROV1
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH PTID
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36
FIXFORM X38
FIXFORM DX2RULE/C1 DX2CODE/5
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH DX2CODE
    ON MATCH REJECT
    ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-*****
CASE PROC8TP
MATCH SITE
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH CLINIC
    ON MATCH CONTINUE
    ON NOMATCH REJECT
MATCH VDATE
    ON MATCH CONTINUE

```

```

      ON NOMATCH REJECT
MATCH PROV1
      ON MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PTID
      ON MATCH CONTINUE
      ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36
FIXFORM X38
FIXFORM ASSESMNT/2
      ON MATCH CONTINUE
      ON NOMATCH REJECT
MATCH ASSESMNT
      ON MATCH REJECT
      ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-*****
CASE PROC9TP
MATCH SITE
      ON MATCH CONTINUE
      ON NOMATCH REJECT
MATCH CLINIC
      ON MATCH CONTINUE
      ON NOMATCH REJECT
MATCH VDATE
      ON MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PROV1
      ON MATCH CONTINUE
      ON NOMATCH REJECT
MATCH PTID
      ON MATCH CONTINUE
      ON NOMATCH REJECT
MATCH LITHO
ON MATCH
FIXFORM X-36
FIXFORM X38
FIXFORM MCODE/6
      ON MATCH CONTINUE
      ON NOMATCH REJECT
MATCH MCODE
      ON MATCH REJECT
      ON NOMATCH INCLUDE
GOTO TOP
ENDCASE
-*****
DATA ON DETRICK
END
TSO FREE F(DETRICK)

```

APPENDIX F

MAINFRAME SAS PROGRAMS (REFERENCED SECTION 6.2.4)

APPENDIX F
MAINFRAME SAS PROGRAMS

TABLE OF CONTENTS

DEERS	DEERS Conversion
PATD32	Patient Social Security Encryption
SIDCONV	SIDPERS Conversion

Note: The listings in this appendix were prepared by downloading the mainframe programs to a PC for printing. In doing so, some IBM EBCDIC characters are not converted to ASCII properly. Two characters may be noticed on the listings. They are the 'not equal' character and a vertical bar used to represent concatenation. The 'not equal' character will be seen as '^='. The vertical bar will be shown as ']'.

```

//DXBP    JOB    (P101),BOLLING,CLASS=R,MSGCLASS=X
//        EXEC  SAS
//DB1     DD     DSN=DXB.REDS.MAY87.DEERS,DISP=SHR
//DB2     DD     DSN=DXB.REDS.MAY87C.DEERS,DISP=OLD
DATA A;
LENGTH LITHO $ 8;
LENGTH Z $ 1;
INFILE DB1 ;
FILE DB2;
INPUT SSN $ 1-9 DEPSEQ $ 11-12 PUIC $ 40-45 ZIP $ 107-111
      BRANCH $ 116 PAYGR $ 117-118 SEX $ 119
      DOB $ 121-126 SPONST $ 127;
UIC='';
IF SPONST = 'A' THEN UIC=PUIC;
IF SPONST = 'B' THEN UIC=PUIC;
IF SPONST = 'E' THEN UIC=PUIC;
IF SPONST = 'J' THEN UIC=PUIC;
IF SPONST = 'N' THEN UIC=PUIC;
IF SPONST = 'V' THEN UIC=PUIC;
IF SPONST = 'Z' THEN UIC=PUIC;
IF SUBSTR(PUIC,1,1) ^= 'W' THEN UIC='';
CAT='X70';
IF SPONST = 'A' THEN
  IF BRANCH = 'A' AND DEPSEQ='20' THEN CAT='A10'; ELSE
  IF BRANCH = 'A' AND DEPSEQ ^= '20' THEN CAT='A50'; ELSE
  IF BRANCH = 'I' AND DEPSEQ='20' THEN CAT='O10'; ELSE
  IF BRANCH = 'I' AND DEPSEQ ^= '20' THEN CAT='O50'; ELSE
  IF BRANCH = 'N' AND DEPSEQ='20' THEN CAT='N10'; ELSE
  IF BRANCH = 'N' AND DEPSEQ ^= '20' THEN CAT='N50'; ELSE
  IF BRANCH = 'M' AND DEPSEQ='20' THEN CAT='M10'; ELSE
  IF BRANCH = 'M' AND DEPSEQ ^= '20' THEN CAT='M50'; ELSE
  IF BRANCH = 'F' AND DEPSEQ='20' THEN CAT='F10'; ELSE
  IF BRANCH = 'F' AND DEPSEQ ^= '20' THEN CAT='F50'; ELSE
  IF BRANCH = 'P' AND DEPSEQ='20' THEN CAT='C10'; ELSE
  IF BRANCH = 'P' AND DEPSEQ ^= '20' THEN CAT='C50'; ELSE
  IF BRANCH = 'E' AND DEPSEQ='20' THEN CAT='P10'; ELSE
  IF BRANCH = 'E' AND DEPSEQ ^= '20' THEN CAT='P50'; ELSE
  CAT='X70';
IF SPONST = 'B' THEN
  IF BRANCH = 'A' AND DEPSEQ='20' THEN CAT='A10'; ELSE
  IF BRANCH = 'A' AND DEPSEQ ^= '20' THEN CAT='A50'; ELSE
  IF BRANCH = 'I' AND DEPSEQ='20' THEN CAT='O10'; ELSE
  IF BRANCH = 'I' AND DEPSEQ ^= '20' THEN CAT='O50'; ELSE
  IF BRANCH = 'N' AND DEPSEQ='20' THEN CAT='N10'; ELSE
  IF BRANCH = 'N' AND DEPSEQ ^= '20' THEN CAT='N50'; ELSE
  IF BRANCH = 'M' AND DEPSEQ='20' THEN CAT='M10'; ELSE
  IF BRANCH = 'M' AND DEPSEQ ^= '20' THEN CAT='M50'; ELSE
  IF BRANCH = 'F' AND DEPSEQ='20' THEN CAT='F10'; ELSE
  IF BRANCH = 'F' AND DEPSEQ ^= '20' THEN CAT='F50'; ELSE
  IF BRANCH = 'P' AND DEPSEQ='20' THEN CAT='C10'; ELSE
  IF BRANCH = 'P' AND DEPSEQ ^= '20' THEN CAT='C50'; ELSE
  IF BRANCH = 'E' AND DEPSEQ='20' THEN CAT='P10'; ELSE

```

```

IF BRANCH ='E' AND DEPSEQ='20' THEN CAT='P50'; ELSE
CAT='X70';
IF SPONST ='N' THEN
IF BRANCH ='A' AND DEPSEQ='20' THEN CAT='A26'; ELSE
IF BRANCH ='A' AND DEPSEQ='20' THEN CAT='A50'; ELSE
IF BRANCH ='F' THEN CAT='F20'; ELSE
CAT='X70';
IF SPONST ='R' THEN
IF BRANCH ='A' AND DEPSEQ='20' THEN CAT='A30'; ELSE
IF BRANCH ='A' AND DEPSEQ='20' THEN CAT='A60'; ELSE
IF BRANCH ='I' AND DEPSEQ='20' THEN CAT='O30'; ELSE
IF BRANCH ='I' AND DEPSEQ='20' THEN CAT='O60'; ELSE
IF BRANCH ='N' AND DEPSEQ='20' THEN CAT='N30'; ELSE
IF BRANCH ='N' AND DEPSEQ='20' THEN CAT='N60'; ELSE
IF BRANCH ='M' AND DEPSEQ='20' THEN CAT='M30'; ELSE
IF BRANCH ='M' AND DEPSEQ='20' THEN CAT='M60'; ELSE
IF BRANCH ='F' AND DEPSEQ='20' THEN CAT='F30'; ELSE
IF BRANCH ='F' AND DEPSEQ='20' THEN CAT='F60'; ELSE
IF BRANCH ='P' AND DEPSEQ='20' THEN CAT='C30'; ELSE
IF BRANCH ='P' AND DEPSEQ='20' THEN CAT='C60'; ELSE
IF BRANCH ='E' AND DEPSEQ='20' THEN CAT='P30'; ELSE
IF BRANCH ='E' AND DEPSEQ='20' THEN CAT='P60'; ELSE
CAT='X70';
IF SPONST ='V' THEN
IF BRANCH ='A' AND DEPSEQ='20' THEN CAT='A23'; ELSE
IF BRANCH ='A' AND DEPSEQ='20' THEN CAT='A50'; ELSE
IF BRANCH ='I' AND DEPSEQ='20' THEN CAT='O20'; ELSE
IF BRANCH ='N' THEN CAT='N20'; ELSE
IF BRANCH ='M' THEN CAT='M20'; ELSE
IF BRANCH ='F' THEN CAT='F20'; ELSE
IF BRANCH ='P' THEN CAT='C20'; ELSE
IF BRANCH ='E' THEN CAT='P20'; ELSE
CAT='X70';
IF SPONST ='J' THEN
IF BRANCH ='A' THEN CAT='A70'; ELSE
IF BRANCH ='N' THEN CAT='N70'; ELSE
IF BRANCH ='F' THEN CAT='F70'; ELSE
IF BRANCH ='P' THEN CAT='C70'; ELSE
CAT='X70';
IF SPONST ='C' THEN CAT='X30';
IF SPONST ='D' THEN CAT='K10';
IF SPONST ='Z' THEN CAT='X70';
IF SPONST ='S' THEN CAT='X20';
IF SPONST ='E' AND DEPSEQ='20' THEN CAT='X20';
XX+1;
LITHO=XX;
DATE='870501';
IF SEX='M' THEN SEX='M' ;
ELSE IF SEX='F' THEN SEX='F';ELSE SEX=' ';
RACE=' ';
PREF=' ';
IF '00'<= PAYGR <= '09' THEN PREF='E ';

```

```

IF '10' <= PAYGR <= '14' THEN PREF='W ' ;
IF '20' <= PAYGR <= '30' THEN PREF='O ' ;
IF '44' <= PAYGR <= '49' THEN PREF='E ' ;
IF '01' <= PAYGR <= '09' THEN PAYGR=PAYGR;
    ELSE IF '11' <= PAYGR <= '14' THEN DO;
        Z=PAYGR-10;
        PAYGR='0']]Z;
    END;
    ELSE IF '21' <= PAYGR <= '29' THEN DO;
        Z=PAYGR-20;
        PAYGR='0']]Z;
    END;
    ELSE IF PAYGR= '30' THEN PAYGR=PAYGR-20;
    ELSE IF '44' <= PAYGR <= '49' THEN DO;
        Z=PAYGR-40;
        PAYGR='0']]Z;
    END;
    ELSE PAYGR=' ' ;
PAD=' ' ;
PAD3=' ' ;
PAD2=' ' ;
PUT SSN $9. DEPSEQ $2. DOB $6. CAT $3. LITHO $8. DATE $6.
SEX $1. RACE $1. PREF $2. PAYGR $2. PAD $4. UIC $6.
PAD3 $3. ZIP $5. PAD2 $12.;

```

```

//DXBS    JOB    (P101),BOLLING,CLASS=R,MSGCLASS=X
//        EXEC  SAS,WORK='90,10'
//DB1     DD     DSN=DXB.BAMC.MAY87C.DEERS,DISP=SHR
//DB2     DD     DSN=DXB.BAMC.MAY87.DPATI,DISP=SHR
PROC FORMAT;
  VALUE CODE 0 ='A' 1 ='B' 2 ='C' 3 ='D' 4 ='E' 5 ='F' 6 ='G'
7 ='H' 8 ='I' 9 ='J' 10 ='K' 11 ='L' 12 ='M' 13 ='N' 14 ='O' 15 ='P'
16 ='Q' 17 ='R' 18 ='S' 19 ='T' 20 ='U' 21 ='V' 22 ='2' 23 ='3'
24 ='4' 25 ='5' 26 ='6' 27 ='7' 28 ='8' 29 ='9' 30 ='0' 31 ='1'
32 =' ' ;
DATA A;
INFILE DB1 ;
FILE DB2;
INPUT SSN 1-9 FMP $CHAR2. OTH $CHAR59.;
SITE='0109';
  K1=INT(SSN/33554432);
  SSN=SSN-K1*33554432;
  K2=INT(SSN/1048576);
  SSN=SSN-K2*1048576;
  K3=INT(SSN/32768);
  SSN=SSN-K3*32768;
  K4=INT(SSN/1024);
  SSN=SSN-K4*1024;
  K5=INT(SSN/32);
  SSN=SSN-K5*32;
  K6=SSN;
PUT SITE $4. K1 CODE1. K2 CODE1. K3 CODE1. K4 CODE1. K5
CODE1. K6 CODE1. FMP $2. OTH $59.;

```

```

//DXBU   JOB   (P101),BOLLING,CLASS=R,MSGCLASS=X
//        EXEC SAS
//DB1    DD    DSN=DXB.POLK.AUG87S.SIDATA,DISP=SHR
//DB2    DD    DSN=DXB.POLK.AUG87.PATI,DISP=OLD
*-----)
*  SIDPERS CONVERSION PROGRAM                               )
*-----);
PROC FORMAT;
  VALUE CODE 0='A' 1='B' 2='C' 3='D' 4='E' 5='F' 6='G'
7='H' 8='I' 9='J' 10='K' 11='L' 12='M' 13='N' 14='O' 15='P'
16='Q' 17='R' 18='S' 19='T' 20='U' 21='V' 22='2' 23='3'
24='4' 25='5' 26='6' 27='7' 28='8' 29='9' 30='0' 31='1'
32=' ';
DATA A;
LENGTH LITHO $ 7;
*-----)
* IMPORTANT: CHANGE SITE CODE (C) AND DATE (YYMMDD)      )
*                                                         )
*   B=BRAGG  C=CAMPBELL  J=JACKSON  P=POLK                )
*                                                         )
*-----);
  C='P';
  DATE='870930';
*-----);
FMP='20';
CAT='A10';
W='W';
PAD1=' ';
PAD12=' ';
PAD3=' ';
INFILE DB1 ;
FILE DB2;
INPUT SSN 1-9 MPC $ 10 GRADE $ 11 SEX $ 12 MOS $CHAR5.
      UPC1 $CHAR5. DOB $CHAR6. RACE $ 29;
      XX+1;
      LITHO=XX;
      *.....;
      ZIP=' ';
      IF C='B' THEN DO; ZIP='28307'; SITE='0089'; END;
      IF C='C' THEN DO; ZIP='42223'; SITE='0060'; END;
      IF C='J' THEN DO; ZIP='29207'; SITE='0105'; END;
      IF C='P' THEN DO; ZIP='71459'; SITE='0064'; END;
      *.....;
      *   CONVERT GRADE                                     .
      *.....;
      IF MPC='E' THEN GOTO ENLIST;
      IF MPC='O' THEN GOTO OFFICE;
      *   ASSUME MPC IS W                                     ;
      IF GRADE='U' THEN DO; GRAD='04'; GOTO GREND; END;
      IF GRADE='V' THEN DO; GRAD='03'; GOTO GREND; END;
      IF GRADE='W' THEN DO; GRAD='02'; GOTO GREND; END;
      IF GRADE='X' THEN DO; GRAD='01'; GOTO GREND; END;

```

```

FILE LOG;
  PUT 'ERROR IN GRADE MPC=' MPC 'GRADE=' GRADE;
FILE DB2;
GRAD='  ';
GOTO GREND;
*.....;
*   GRADE CONVERSION WHEN MPC=E
ENLIST: IF GRADE='M' THEN DO; GRAD='04'; GOTO GREND; END;
        IF GRADE='N' THEN DO; GRAD='05'; GOTO GREND; END;
        IF GRADE='O' THEN DO; GRAD='06'; GOTO GREND; END;
        IF GRADE='P' THEN DO; GRAD='07'; GOTO GREND; END;
        IF GRADE='R' THEN DO; GRAD='09'; GOTO GREND; END;
        IF GRADE='X' THEN DO; GRAD='07'; GOTO GREND; END;
        IF GRADE='Y' THEN DO; GRAD='08'; GOTO GREND; END;
        IF GRADE='1' THEN DO; GRAD='01'; GOTO GREND; END;
        IF GRADE='2' THEN DO; GRAD='02'; GOTO GREND; END;
        IF GRADE='3' THEN DO; GRAD='03'; GOTO GREND; END;
        IF GRADE='4' THEN DO; GRAD='04'; GOTO GREND; END;
        IF GRADE='5' THEN DO; GRAD='05'; GOTO GREND; END;
        IF GRADE='6' THEN DO; GRAD='06'; GOTO GREND; END;
        IF GRADE='7' THEN DO; GRAD='07'; GOTO GREND; END;
        IF GRADE='8' THEN DO; GRAD='08'; GOTO GREND; END;
        IF GRADE='9' THEN DO; GRAD='09'; GOTO GREND; END;

FILE LOG;
  PUT 'ERROR IN GRADE MPC=' MPC 'GRADE=' GRADE;
FILE DB2;
GRAD='  ';
GOTO GREND;
*.....;
*   GRADE CONVERSION WHEN MPC=O
OFFICE: IF GRADE='A' THEN DO; GRAD='08'; GOTO GREND; END;
        IF GRADE='B' THEN DO; GRAD='06'; GOTO GREND; END;
        IF GRADE='C' THEN DO; GRAD='05'; GOTO GREND; END;
        IF GRADE='D' THEN DO; GRAD='04'; GOTO GREND; END;
        IF GRADE='E' THEN DO; GRAD='03'; GOTO GREND; END;
        IF GRADE='F' THEN DO; GRAD='02'; GOTO GREND; END;
        IF GRADE='G' THEN DO; GRAD='01'; GOTO GREND; END;
        IF GRADE='5' THEN DO; GRAD='03'; GOTO GREND; END;
        IF GRADE='6' THEN DO; GRAD='02'; GOTO GREND; END;
        IF GRADE='7' THEN DO; GRAD='01'; GOTO GREND; END;

FILE LOG;
  PUT 'ERROR IN GRADE MPC=' MPC 'GRADE=' GRADE;
FILE DB2;
GRAD='  ';
*.....;
*   CONVERT RACE
GREND: IF RACE=' ' THEN RACE='7';
        IF RACE='C' THEN RACE='5';
        IF RACE='M' THEN RACE='2';
        IF RACE='N' THEN RACE='4';
        IF RACE='R' THEN RACE='1';
        IF RACE='X' THEN RACE='7';

```

```

*.....;
*   CONVERT JOB CODE
      IF MPC='E' THEN JC='0']]SUBSTR(MOS,1,3);
      IF MPC='O' THEN JC='0']]SUBSTR(MOS,1,3);
      IF MPC='W' THEN JC=SUBSTR(MOS,1,3)]]' ';
*.....;
*   CONVERT SSN TO BASE 32 ENCRYPTION
      K1=INT(SSN/33554432);
      SSN=SSN-K1*33554432;
      K2=INT(SSN/1048576);
      SSN=SSN-K2*1048576;
      K3=INT(SSN/32768);
      SSN=SSN-K3*32768;
      K4=INT(SSN/1024);
      SSN=SSN-K4*1024;
      K5=INT(SSN/32);
      SSN=SSN-K5*32;
      K6=SSN;
*.....;
PUT SITE $4. K1 CODE1. K2 CODE1. K3 CODE1. K4 CODE1. K5
CODE1. K6 CODE1. FMP $2. DOB $6. CAT$3. LITHO $7. C $1.
DATE $6. SEX $1. RACE $1. MPC $1. PAD1 $1. GRAD $2. JC $4.
W $1. UPC1 $5. PAD3 $3. ZIP $5. PAD12 $12.;

```

Offset	Data	Source Line	IBM Personal Computer BASIC Compiler V1.00
001A	0002	1 REM INPUT PARAMETERS	
001A	0002	PRINT "ENTER INPUT FILE PRECEDED BY DISK DESIGNATION ";	
0037	0002	INPUT INPF\$	
0047	0006	PRINT "ENTER OUTPUT FILE PRECEDED BY DISK DESIGNATION ";	
004F	0006	INPUT OUTF\$	
005F	000A	PRINT "ENTER STARTING SEQUENCE NUMBER ";	
0067	000A	INPUT SEQ	
0077	000E	PRINT "ENTER SITE CODE"	
007F	000E	PRINT "USE FIRST B=BRAGG C=CAMPBELL J=JACKSON P=POLK ";	
0087	000E	INPUT CS	
0097	0012	PRINT "ENTER CURRENT DATE YYMMDD ";	
009F	0012	INPUT DAT\$	
00AF	0016	CLS	
00B3	0016	PRINT "INPUT FILE ";INPF\$	
00C0	0016	PRINT "OUTPUT FILE ";OUTF\$	
00CD	0016	PRINT "STARTING SEQ NO. ";SEQ	
00DA	0016	PRINT "SITE CODE ";CS	
00E7	0016	PRINT "CURRENT DATE ";DAT\$	
00F4	0016	PRINT	
00FC	0016	PRINT "VERIFY ABOVE. CONTINUE? (Y/N) ";	
0104	0016	INPUT ANSS	
0114	001A	IF ANSS<>"Y" THEN GOTO 1	
0126	001A	OPEN INPF\$ FOR INPUT AS #1	
0137	001A	OPEN OUTF\$ FOR OUTPUT AS #2	
0149	001A	PROGID\$="200"	
0152	001E	FMP\$="20"	
0158	0022	CAT\$="A10"	
0164	0026	W\$="W"	
016D	002A	FILE\$="<-----"	
0176	002E	IF ASC(C\$) > 91 THEN C\$=CHR\$(ASC(C\$)-32)	
0194	002E	ZIP\$=" "	
019D	0032	IF C\$="B" THEN ZIP\$="28307"	
01BA	0032	IF C\$="C" THEN ZIP\$="42223"	
01CB	0032	IF C\$="J" THEN ZIP\$="29207"	
01E2	0032	IF C\$="P" THEN ZIP\$="71459"	
01F9	0032		
01F9	0032	SEQ=SEQ-1	
0206	0032		
0206	0032	100 REM READ RECORD	
0207	0032		
0207	0032	IF EOF(1) THEN GOTO 500	
0219	0032	INPUT #1,A\$	
0228	0036	SEQ = SEQ + 1	
0235	0036	SSN\$ = LEFT\$(A\$,9)	
0244	003A	MPC\$ = MID\$(A\$,10,1)	
0256	003E	GRADE\$ = MID\$(A\$,11,1)	
0268	0042	SEX\$ = MID\$(A\$,12,1)	
027A	0046	DWOSS\$ = MID\$(A\$,13,5)	
028C	004A	UPC1\$ = MID\$(A\$,18,5)	
029E	004E	DOB\$ = MID\$(A\$,23,6)	
02B0	0052	RACE\$ = MID\$(A\$,29,1)	
02C2	0056	IF LEN(DOB\$)=0 THEN DOB\$=" "	
02DB	0056	IF LEN(RACE\$)=0 THEN RACE\$=" "	
02EF	0056		

Offset Data Source Line IBM Personal Computer BASIC Compiler V1.00

```

02EE 0056 REM CONVERT SEQ NO.
02EE 0056 SEQ$ = STR$(SEQ)
02FA 005A N = LEN(SEQ$)
0307 005E SEQ$ = STRING$(8-N,"0") + RIGHT$(SEQ$,N-1)
0336 005E
0336 005E REM CONVERT GRADE
0336 005E IF MPC$="E" THEN GOTO 120
0348 005E IF MPC$="O" THEN GOTO 140
035A 005E REM ASSUME MPC IS W
035A 005E IF GRADE$="U" THEN GRADE$="04" :GOTO 160
0375 005E IF GRADE$="V" THEN GRADE$="03" :GOTO 160
0390 005E IF GRADE$="W" THEN GRADE$="02" :GOTO 160
03AB 005E IF GRADE$="X" THEN GRADE$="01" :GOTO 160
03C6 005E PRINT "ERROR IN GRADE MPC=";MPC$,"GRADE=";GRADE$
03D0 005E GRADE$=" "
03E6 005E GOTO 160
03EA 005E
03EA 005E 120 REM GRADE CONVERSION FOR MPC=E
03EB 005E IF GRADE$="M" THEN GRADE$="04" :GOTO 160
0406 005E IF GRADE$="N" THEN GRADE$="05" :GOTO 160
0421 005E IF GRADE$="O" THEN GRADE$="06" :GOTO 160
043C 005E IF GRADE$="P" THEN GRADE$="07" :GOTO 160
0457 005E IF GRADE$="R" THEN GRADE$="09" :GOTO 160
0472 005E IF GRADE$="X" THEN GRADE$="07" :GOTO 160
048D 005E IF GRADE$="Y" THEN GRADE$="08" :GOTO 160
04A8 005E IF GRADE$="1" THEN GRADE$="01" :GOTO 160
04C3 005E IF GRADE$="2" THEN GRADE$="02" :GOTO 160
04DE 005E IF GRADE$="3" THEN GRADE$="03" :GOTO 160
04F9 005E IF GRADE$="4" THEN GRADE$="04" :GOTO 160
0514 005E IF GRADE$="5" THEN GRADE$="05" :GOTO 160
052F 005E IF GRADE$="6" THEN GRADE$="06" :GOTO 160
054A 005E IF GRADE$="7" THEN GRADE$="07" :GOTO 160
0565 005E IF GRADE$="8" THEN GRADE$="08" :GOTO 160
0580 005E IF GRADE$="9" THEN GRADE$="09" :GOTO 160
0598 005E PRINT "ERROR IN GRADE MPC=";MPC$,"GRADE=";GRADE$
05B2 005E GRADE$=" "
05BB 005E GOTO 160
05BF 005E
05BF 005E 140 REM GRADE CONVERSION WHEN MPC=O
05C0 005E IF GRADE$="A" THEN GRADE$="08" :GOTO 160
05DB 005E IF GRADE$="B" THEN GRADE$="06" :GOTO 160
05F6 005E IF GRADE$="C" THEN GRADE$="05" :GOTO 160
0611 005E IF GRADE$="D" THEN GRADE$="04" :GOTO 160
062C 005E IF GRADE$="E" THEN GRADE$="03" :GOTO 160
0647 005E IF GRADE$="F" THEN GRADE$="02" :GOTO 160
0662 005E IF GRADE$="G" THEN GRADE$="01" :GOTO 160
067D 005E IF GRADE$="5" THEN GRADE$="03" :GOTO 160
0698 005E IF GRADE$="6" THEN GRADE$="02" :GOTO 160
06B3 005E IF GRADE$="7" THEN GRADE$="01" :GOTO 160
06CE 005E PRINT "ERROR IN GRADE MPC=";MPC$,"GRADE=";GRADE$
06E5 005E GRADE$=" "
06EE 005E
06EE 005E 160 REM END OF GRADE CONVERSION
06EF 005E

```

Offset Data Source Line IBM Personal Computer BASIC Compiler V1.00

```

06EF 005E REM CONVERT RACE
06EF 005E IF RACE$=" " THEN RACE$="7"
0706 005E IF RACE$="C" THEN RACE$="5"
071D 005E IF RACE$="H" THEN RACE$="2"
0734 005E IF RACE$="N" THEN RACE$="4"
074B 005E IF RACE$="R" THEN RACE$="1"
0762 005E IF RACE$="X" THEN RACE$="7"
0779 005E
0779 005E REM CONVERT JOBCODE
0779 005E JCS = LEFT$(DWOSS,3)
0788 0062 IF MPC$="E" THEN JCS= "0"+JCS
07A3 0062 IF MPC$="O" THEN JCS= "0"+JCS
07BE 0062 IF MPC$="W" THEN JCS= JCS+" "
07D8 0062
07D8 0062 REM BUILD OUTPUT RECORD
07D8 0062 RECS=PROGID$ + SSNS + FMP$ + DOBS + CAT$ + SEQNS + CS
0804 0066 RECS=RECS + DAT$ + SEX$ + RACE$ + MPC$ + " " + GRADES
082F 0066 RECS=RECS + JCS + WS + UPC1$ + " " + ZIP$ + " "
085A 0066 RECS=RECS + FIL$
0866 0066
0866 0066 PRINT #2,RECS
0871 0066 GOTO 100
0875 0066
0875 0066 500 REM CLOSE FILES AND QUIT
0876 0066 CLOSE 1,2
0883 0066 END
0887 0066
088A 0066

```

22151 Bytes Available

19949 Bytes Free

0 Warning Error(s)

0 Severe Error(s)